مكتبة بروكيـــن سوفــت

BROKENSOFT.COM



كتب أجنبيه

كتب البرمجه

كتب المنتديات

كتب شرح البرامج

كتب تطوير المواقع

كتب برمجــة الإنترنت

كتب قواعــد البيانـــات

كتب التصميو والجرافيكس

كتب و مختارات من المنتديات

مكتبة بروكين سوفت

إحدى أسلحة شبكة النينجا



أنشأ مشروعك بالفيجوال بيسك ٦
من الألف الى الياء

أعداد

ناصر محمد سالم

Nasser50001

الإهداء

إلى أخواني وأخواتي أعضاء منتديات الفريق العربي للبرمجة الذين لهم في القلب مكانة كبيرة حيث يعمل الجميع من اجل الجميع ويعلم الجميع الجميع ولا يبخل أحد بما لدية من علم على أحد حيث من النادر أن تجد هذا الجو الأخوي الجميل في غير هذه الأماكن ....

إلى هؤلاء الصفوة من المبرمجين أهدي هذا الكتاب وأسأل الله تعالى أن تعم الاستفادة منه وان يكون صدقة جارية إلى يوم القيامة

أخوكم

ناصر محمد سالم

Nasser50001

السعر الخاص بالنسخة

أخي الكريم قارئ هذا الكتاب ....

إن أحسست أنك استفدت من هذا الكتاب شيئا أو قدم لك معلومة فالمطلوب منك شيئين

الأول : الدعاء لصاحب الكتاب بظاهر الغيب

الثاني : أن تتبرع بما قيمته واحد ريال لأهلنا المسلمين في فلسطين أو العراق أو أي بقعة من بقاع الأرض

وهكذا يصبح الكتاب ملك لك

الفهرس

الحلقة الأولى

الخطوة الأولى : تحليل النظام

قبل البدء في أي برنامج (بالطبع سيكون تركيزي على برامج قواعد البيانات) يجب عمل تحليل للبرنامج ومن ثم نقوم بتصميم قاعدة البيانات المطلوبة ثم البدء بالعمل مع البرنامج وأول خطوة في تحليل البرنامج هي

١ - التعرف على وظائف النظام

ويتم التعرف على وظائف النظام عن طريق جمع المعلومات الدقيقة وأخذ صورة من كل التقارير اليدوية والعمليات التي كانت تتم في النظام اليدوي وكذلك الجلوس جلسات مطولة مع المسئولين والمديرين ومديري الأقسام والموظفين لمعرفة خوارزمية النظام بالكامل وتكون على اتصال معهم والسؤال عن كل نقطة داخل النظام وما هي وجهة نظرهم في النظام وكيفية الشاشات وطرقة طباعة التقارير ....

وكمثال عملي برنامج عن مؤسسة تجارية لبيع سلعة (الصافي – المراعي .... مثلا) حيث المفترض في النظام أن يقوم بالعمليات التالية :

- طباعة تقرير عن بيانات عملاء المؤسسة
- طباعة تقرير عن بيانات مندوبي المبيعات
- طباعة تقرير عن بيانات الموردين (إن كان هناك موردين للمؤسسة)
- طباعة تقرير عن بيانات الأصناف .
- تقرير بجميع الطلبيات لعميل معين
- تقرير بجميع الطلبيات لمندوب معين
- تقرير بجميع الطلبيات بتاريخ معين أو بين تاريخين أو يومية ....

٢ – تصميم قاعدة البيانات

من النقطة ١ أمكننا تحديد العناصر الرئيسية في النظام حيث يمكننا البدء بتصميم قاعدة البيانات كالتالي :

١ – **جدول بيانات العملاء** : حيث يوضع فيه حقول تعبر عن جميع بيانات العملاء المطلوبة منك (رقم العميل – اسم العميل – العنوان – الهاتف – الموبايل – البريد الإلكتروني – مجال

العمل ...... وأي بيانات أخرى تريد إضافتها أنشأ لها حقل مع تحديد نوع البيانات لكل حقل وحجم كل حقل .

٢ – جدول بيانات المندوبين : حيث يوضع فيه حقول تعبر عن جميع بيانات المندوبين المطلوبة منك (رقم المندوب – اسم المندوب – العنوان – الهاتف – الموبايل – البريد الإلكتروني – المنطقة - ...... وأي بيانات أخرى تريد إضافتها أنشأ لها حقل مع تحديد نوع البيانات لكل حقل وحجم كل حقل وهكذا في كل جدول

ملاحظات هامة جدا عند تصميم الجدول

١ – أن جميع الحقول يجب أن تكون مرتبطة بحالة واحدة (العملاء مثلا)

٢ - إذا حدث تكرار للبيانات داخل الجدول فيجب فصل الجدول إلى جدولين

٣ – أن جميع الحقول تكون مرتبطة بالمفتاح الأساسي للجدول (رقم العميل مثلا)

هذه أهم الملاحظات على تصميم الجداول .

تشكيل البيانات Data Normalization

هي مجموعة من الخطوات لإعادة تشكيل الجداول بطريقة صحيحة حتى لا يحدث بين البيانات تعارض داخل الجداول وهي تسير في ثلاث أشكال رئيسية

الشكل الأول للبيانات

حيث يتم في هذه الخطوة إلغاء كافة الحقول المكررة داخل الجدول فإذا كان عندنا الجدول الخاص بالفاتورة للطلبيات كالتالي :

رقم الفاتورة	تاريخ الفاتورة	صنف ١	صنف٢	صنف ٣	الإجمالي
١٥٢	٢٠٠٤/٨/٥	Mouse	keyboard	Floppy	

فإن الصنف في الجدول الأول تكرر أكثر من مرة وتصبح هناك مشكلة كبيرة إذا كان عدد الأصناف كبير ولحل هذه المشكلة يتم حذف الحقول المكررة بوضع حقل بديل لمجموعة الحقول مثل حقل رقم الصنف كبديل عن مجموعة الحقول ويصبح الجدول كالتالي :

رقم الفاتورة	تاريخ الفاتورة	رقم الصنف	الإجمالي
١٥٢	٢٠٠٤/٨/٥	**Mouse**	
١٥٢	٢٠٠٤/٨/٥	**Keyboard**	
١٥٢	٢٠٠٤/٨/٥	**Floppy**	

وهذا هو الشكل الأول للبيانات

الشكل الثاني للبيانات

كل حقل من حقول الجدول يجب أن يكون مرتبط (ارتباط معلومات) بالمفتاح الأساسي للجدول وأي حقول لا ترتبط بالحقل الأساسي يتم فصلها في جدول مستقل .

حيث يعرف المفتاح الأساسي للجدول أنه حقل رقمي ترتبط به كل حقول الجدول ويستخدم في عملية الفرز والبحث بحيث أنه حقل وحيد داخل الجدول

ففي الجدول الذي تم استخراجه في الشكل الأول نلاحظ أن رقم المنتج غير مرتبط بالحقل الأساسي للجدول وهو رقم الفاتورة لذلك نجد وجود تكرار في بيانات الحقل الأول رقم الفاتورة ولذلك يتم قسم الجدول إلى جدولين أحدهما لبيانات الفاتورة والآخر لبيانات الأصناف ويجب عمل حساب أن الجدولين يجب أن يتم الربط بينهما

الجدول الأول

رقم الفاتورة	تاريخ الفاتورة	الإجمالي
١٥٢	٢٠٠٤/٨/٥	١٥٠٠
١٥٣	٢٠٠٤/٨/٦	٢٥٠٠
١٥٤	٢٠٠٤/٨/٧	١٦٠٠

الجدول الثاني

رقم الفاتورة	رقم الصنف	اسم الصنف	سعر البيع	سعر الشراء	تاريخ الصلاحية
١٥٢	١	**Mouse**			
١٥٢	٢	**Keyboard**			
١٥٢	٣	**Floppy**			

الشكل الثالث للبيانات

كل حقل بالجدول يعتمد اعتمادا وحيدا على المفتاح الأساسي للجدول فمـثلا إذا كـان عنـدنا الجدول

رقم الطالب	رمز المادة	اسم المادة	درجة المادة
١٥٢	١	**Mouse**	
١٥٢	٢	**Keyboard**	

فإن الحقل اسم المادة غير مرتبط ارتباطا وحيدا برقم الطالب

فيتم تقسيم الجدول إلى جدولين كالتالي :

الجدول الأول

رقم الطالب	رمز المادة	درجة المادة
١٥٢	١٠١حاب	١٠

الجدول الثاني

رمز المادة	اسم المادة
١٠١حاب	مقدمة حاسب

وهكذا يتم تشكيل البيانات لكل الجدول لداخل قاعدة البيانات المستخدمة وخذ الوقت الكافي لذلك حتى يمكنك تصميم قاعدة بيانات جيدة لا يحدث بها مشاكل عند كتابة البرمجة للبرنامج بعد ذلك بعد تصميم قاعدة البيانات يتم إنشاء قاعدة البيانات باستخدام برنامج قاعدة البيانات (أكسس أو SERVER SQL إذا كنت تود استخدام البرنامج داخل شبكة محلية أو شبكة موسعة

الحلقة الثانية

الخطوة الثانية : تصميم شاشة البرنامج

تصميم الشاشة الرئيسية للبرنامج وهي واجهة البرنامج التي ستكون ساحة العمل ونريد أن نصمم شاشة رئيسية كالشاشة التالية :



إنشاء القوائم وأشرطة الأدوات والشريط الحاضن

أولا : إنشاء القوائم

لإنشاء القوائم : افتح قائمة TOOLS واختار منها MENU EDITOR أو اضغط مفتاحي CTRL + E أو أضغط الأداة في شريط الأدوات تظهر الشاة التالية



ويمكن تقسيمها إلى الأجزاء التالية

(١)



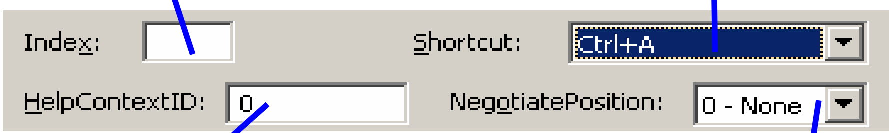
كتابة عنوان القائمة الذي سيظهر على النموذج

كتابة اسم القائمة الذي سيستخدم في الكود

(٢)

تستخدم إذا أردت إنشاء مصفوفة قوائم

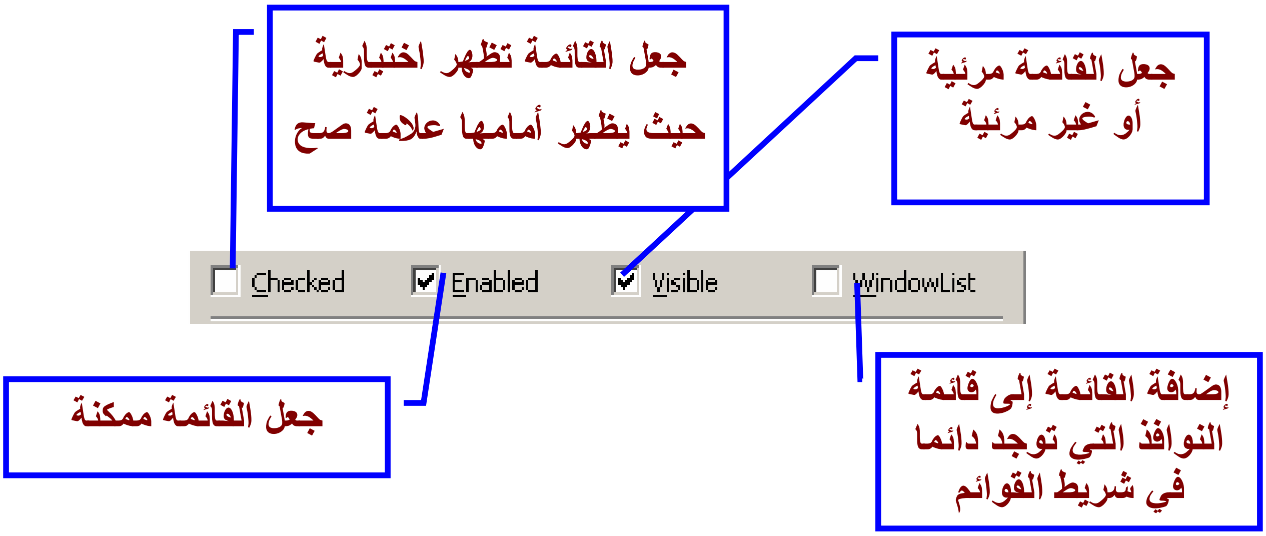
تحديد مجموعة مفاتيح لفتح القائمة بدلا من الضغط على القائمة بالماوس



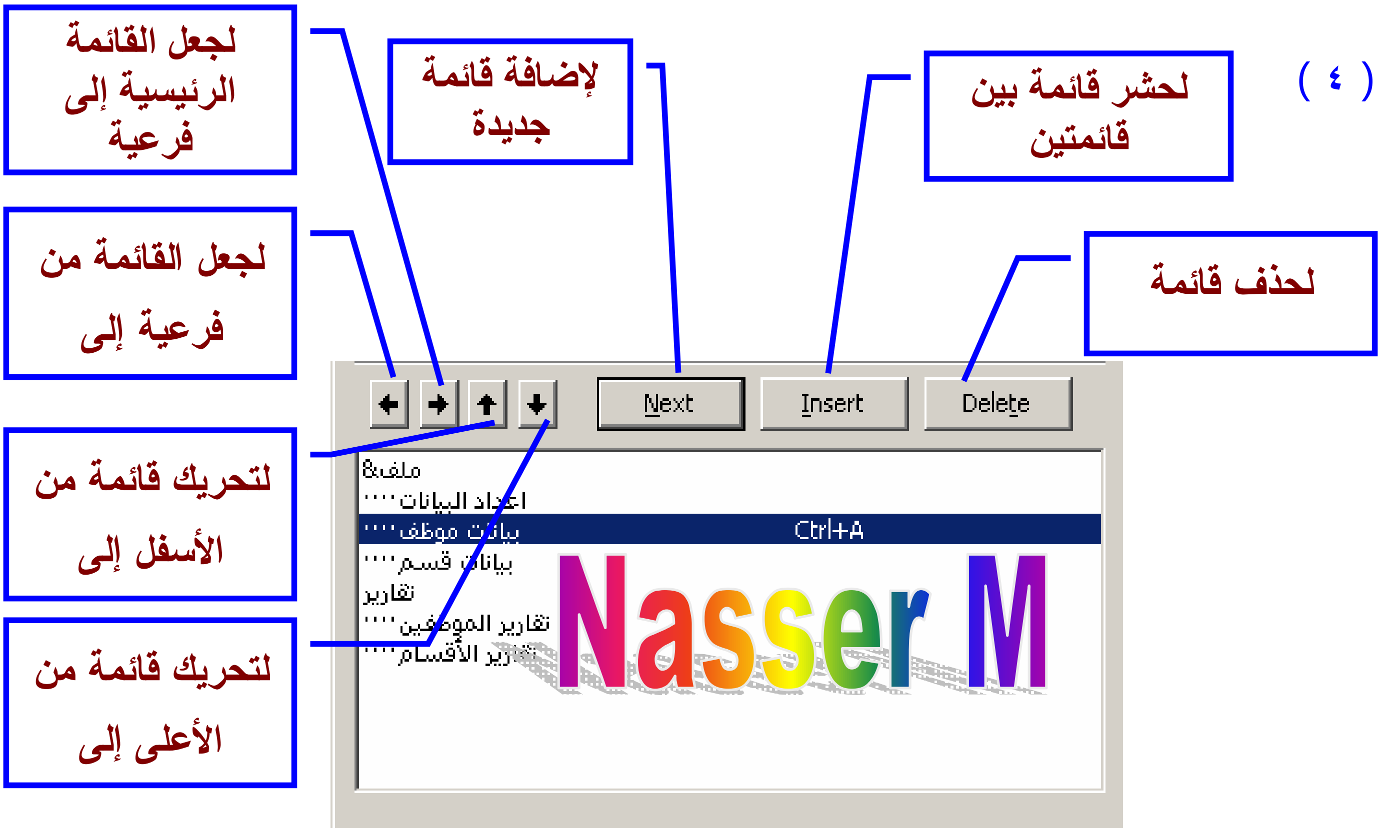
يكتب هنا رقم ملف التعليمات المراد عرضه

تحديد موضع ظهور القائمة ولها ثلاث خيارات يمين ، يسار ، ووسط

(٣)



(٤)

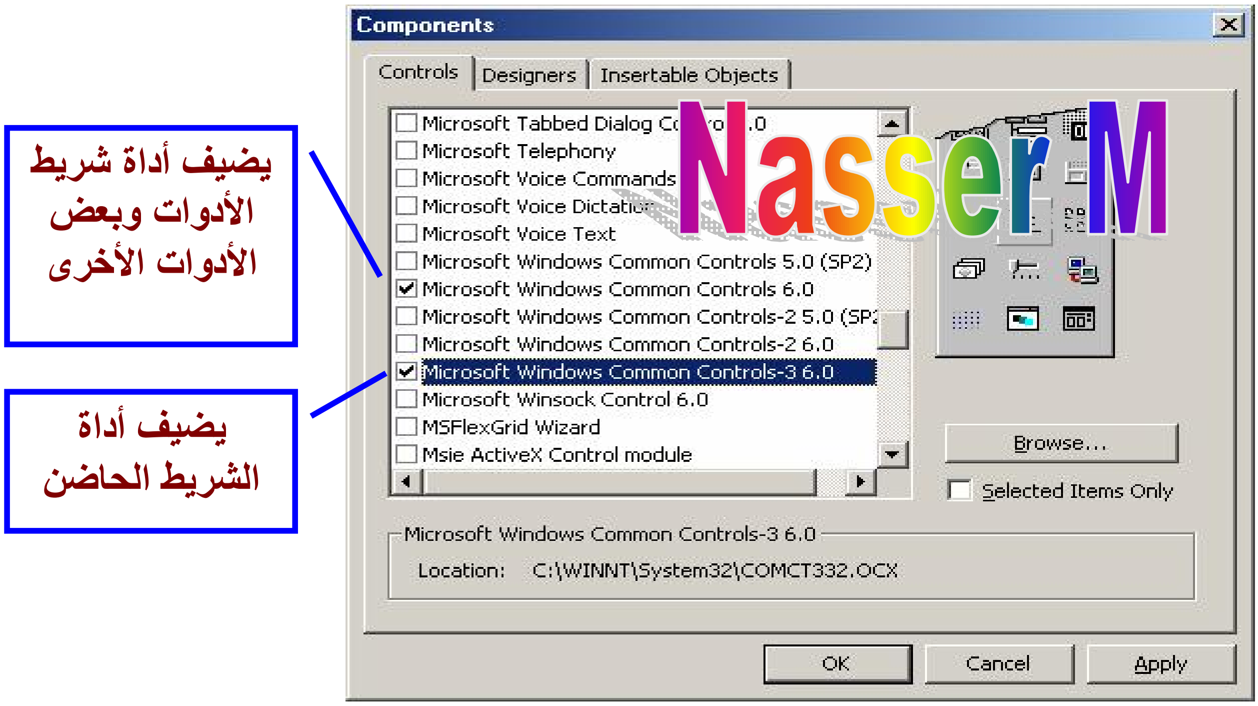


وبهذا تم إنشاء القوائم ويمكن جعلها جهة اليمين أو اليسار من خاصية Right To Left من خصائص النموذج Form

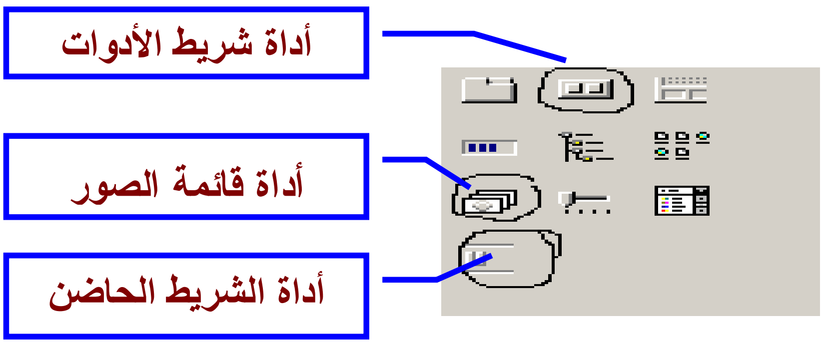
ثانيا إنشاء أشرطة الأدوات و الشريط الحاضن

١ – يجب أولا لإضافة هذه المجموعة من الأدوات إلى شريط الأدوات كالتالي

افتح قائمة **PROJECT** ثم اختار **COMPONENTS** تظهر الشاشة التالية حدد الاختيار الذي عليه سهم

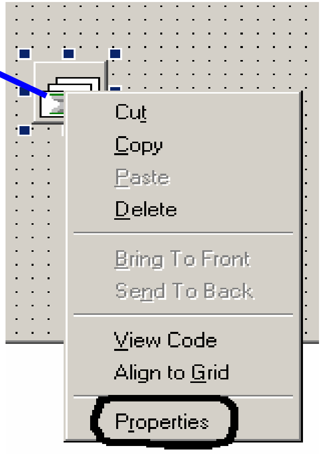


فتظهر مجموعة من الأدوات الجديدة في شريط الأدوات كما هو موضح في الصورة

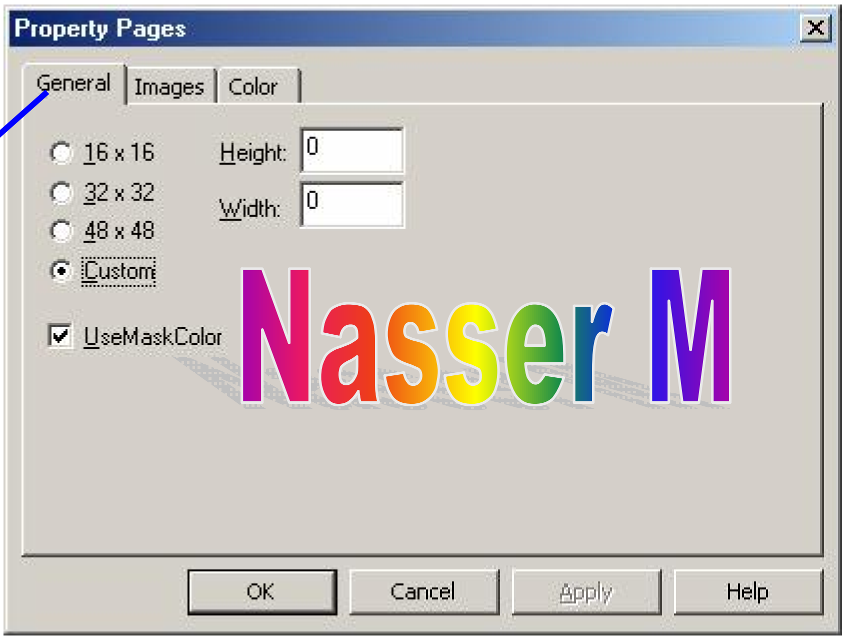


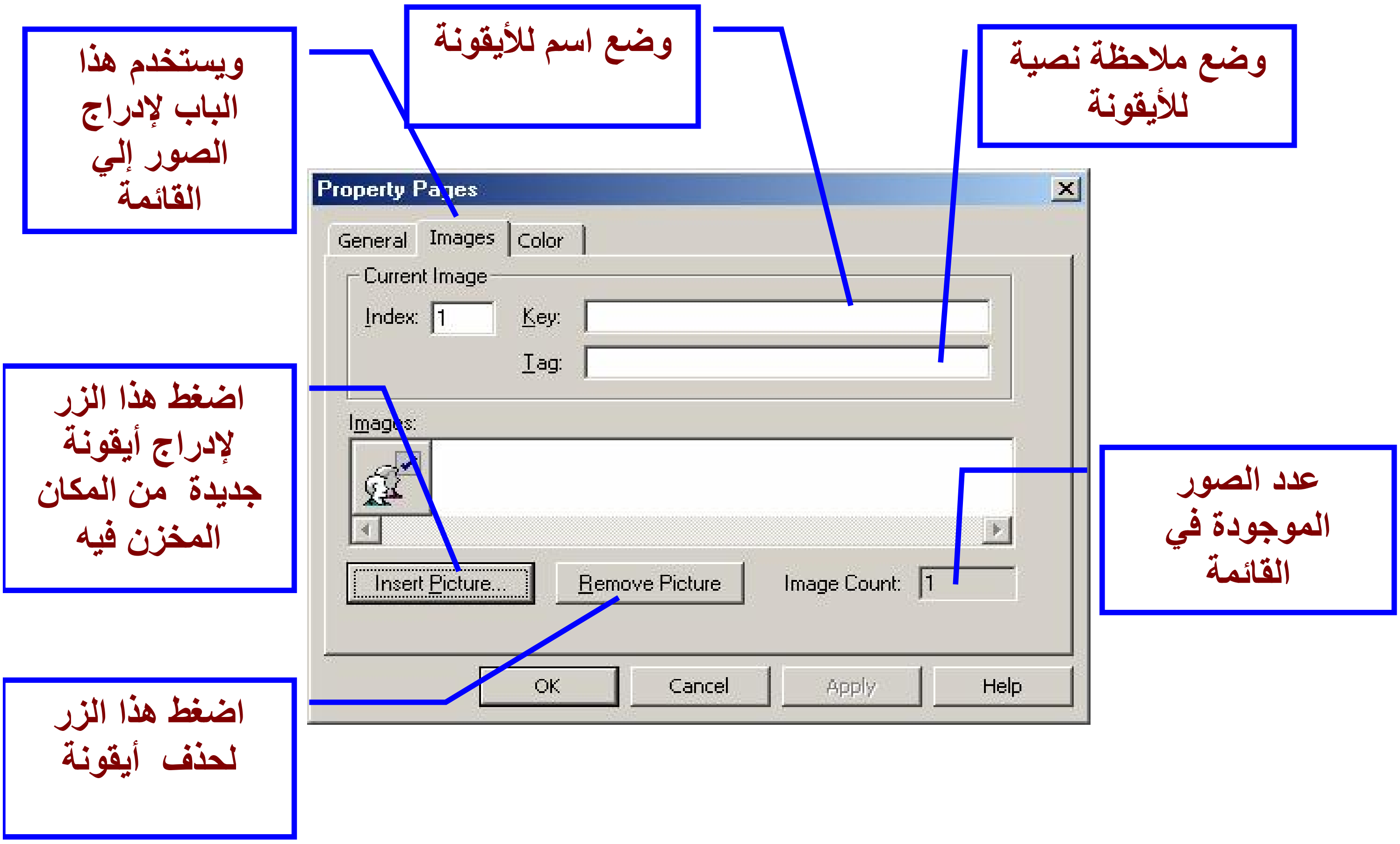
أولا : نضع أداة قائمة الصور لوضع الأيقونات الخاصة بالأزرار على شريط الأدوات كما هو موضح بالصور

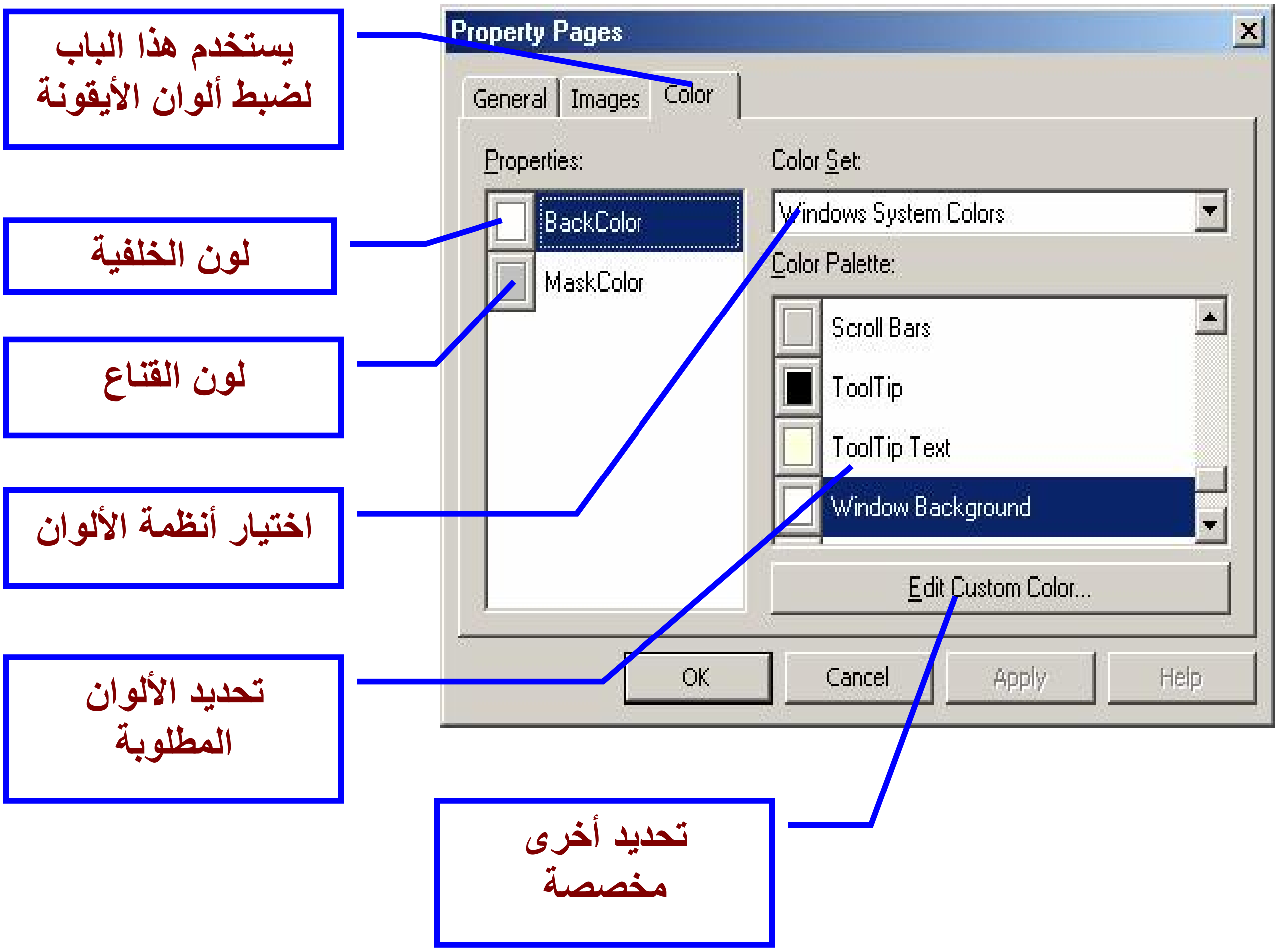
اضغط على الأداة بالزر
الأيمن واختار خصائص



يستخدم هذا الباب لتحديد
حجم الصورة التي تظهر
على الشريط حدد
١٦x١٦ويمكنك تحديد
العرض والارتفاع من
width و height




ويستخدم هذا الباب لإدراج الصور إلي القائمة
وضع اسم للأيقونة
وضع ملاحظة نصية للأيقونة
Property Pages
General
Images
Color
Current Image
Index:
1
Key:
Tag:
Images:
Insert Picture...
Remove Picture
Image Count:
1
OK
Cancel
Apply
Help
اضغط هذا الزر لإدراج أيقونة جديدة من المكان المخزن فيه
عدد الصور الموجودة في القائمة
اضغط هذا الزر لحذف أيقونة


يستخدم هذا الباب لضبط ألوان الأيقونة
Property Pages
General
Images
Color
Properties:
Color Set:
BackColor
MaskColor
Windows System Colors
Color Palette:
Scroll Bars
ToolTip
ToolTip Text
Window Background
Edit Custom Color...
OK
Cancel
Apply
Help
لون الخلفية
لون القناع
اختيار أنظمة الألوان
تحديد الألوان المطلوبة
تحديد أخرى مخصصة

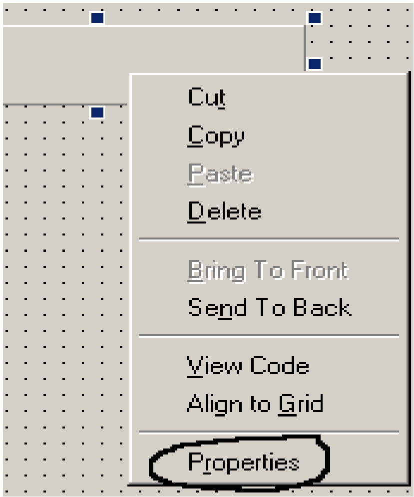
إنشاء أشرطة الأدوات

نرسم شريط الأدوات على الشاشة ثم نضبط خصائص شريط الأدوات التالية

القيمة	الخاصية
Tb1	name
0 vbalignnon	Align المحاذاة

ثم اسحب شريط الأدوات إلى الأسفل وصغره كما في الصورة

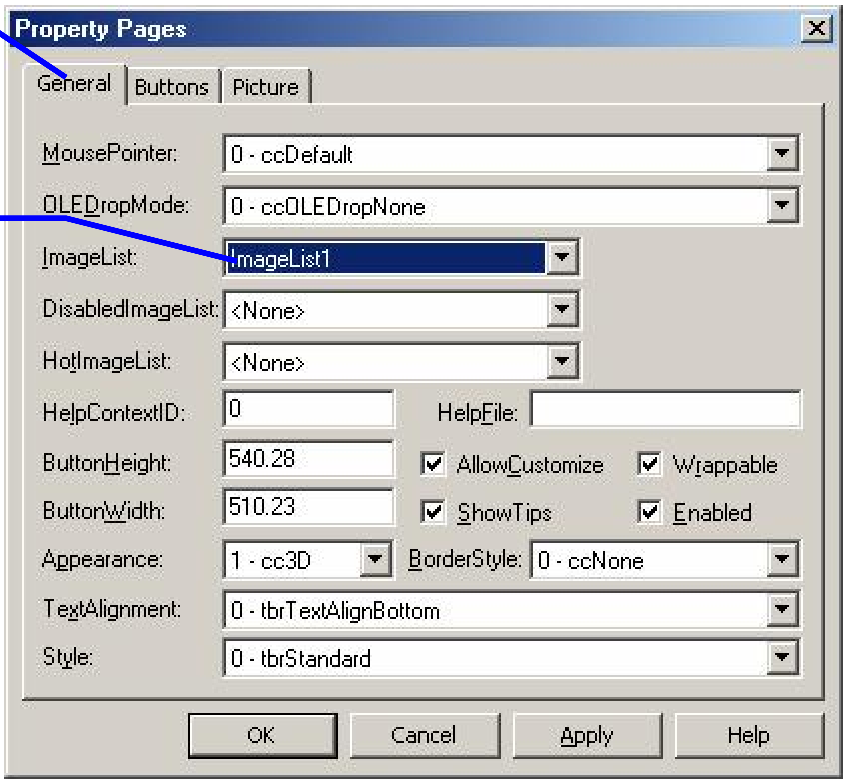
اضغط على الشريط بالزر الأيمن – ثم خصائص كما في الصورة



تظهر الشاشة التالية

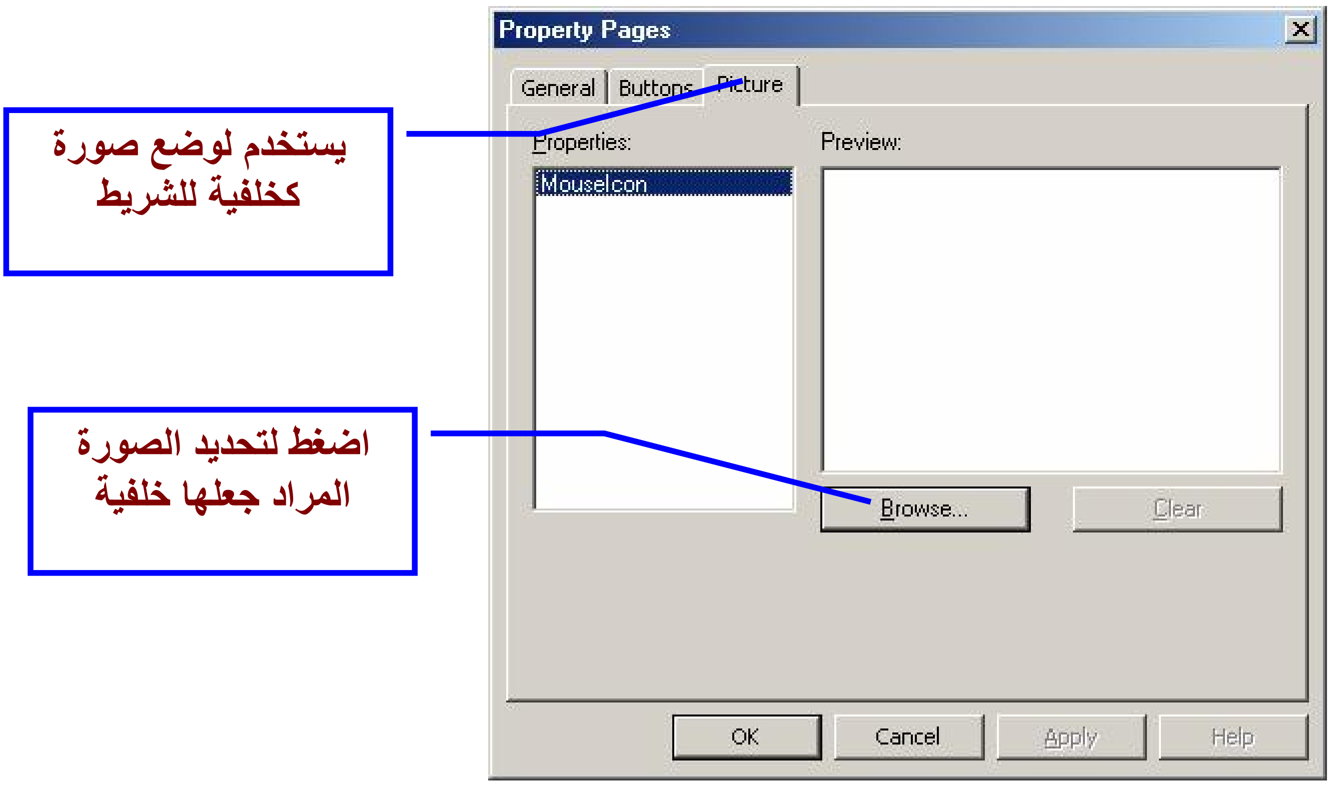
هذا الباب لإعداد خصائص الشريط

لاختيار قائمة الصور التي يتم ربطها مع الشريط بحيث سيتم وضع صورة على كل أداة من ادوات الشريط





الباب الثالث باب الصورة



كرر نفس الخطوات السابقة لوضع شريط آخر

إنشاء الشريط الحاضن

يستخدم الشريط الحاضن في احتواء أشرطة الأدوات مثلما يظهر في شريط أدوات مستعرض الإنترنت ضع الشريط  على النموذج واضبط خصائصه كالتالي

الخاصية	القيمة
name	cb
Align	1 vbaligntop

لربط أشرطة الأدوات مع الشريط الحاضن يوجد طريقتين :

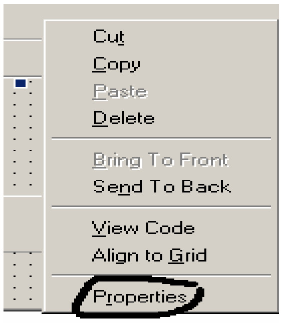
الطريقة الأولى : من خصائص الشريط الحاضن

اضغط على شريط الأدوات بالزر الأيمن وأختار cut ثم أضغط على اللوح المراد وضع الشريط عليه بالزر الأيمن وأختار paste

وهكذا أكرر مع كل شريط أدوات أريد وضعه في الشريط الحاضن

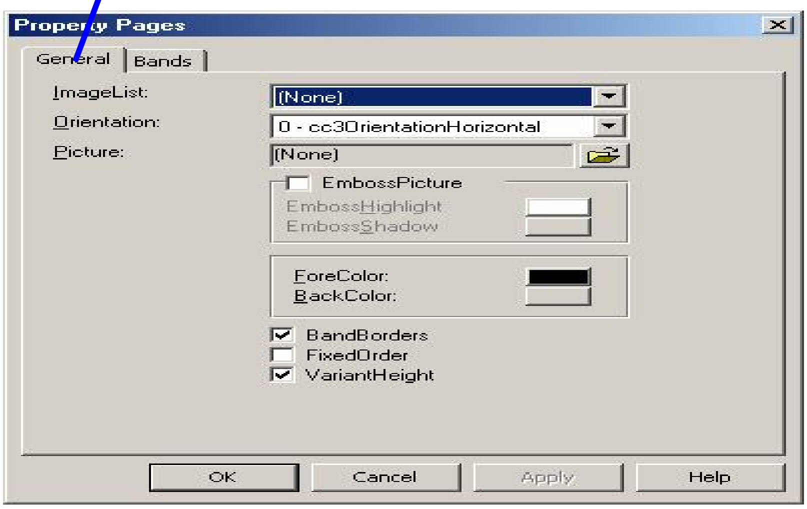
ضبط خصائص الشريط الحاضن

اضغط على الشريط الحاضن بالزر الأيمن وأختار خصائص كما في الصورة



لضبط الخصائص العامة للشريط الحاضن

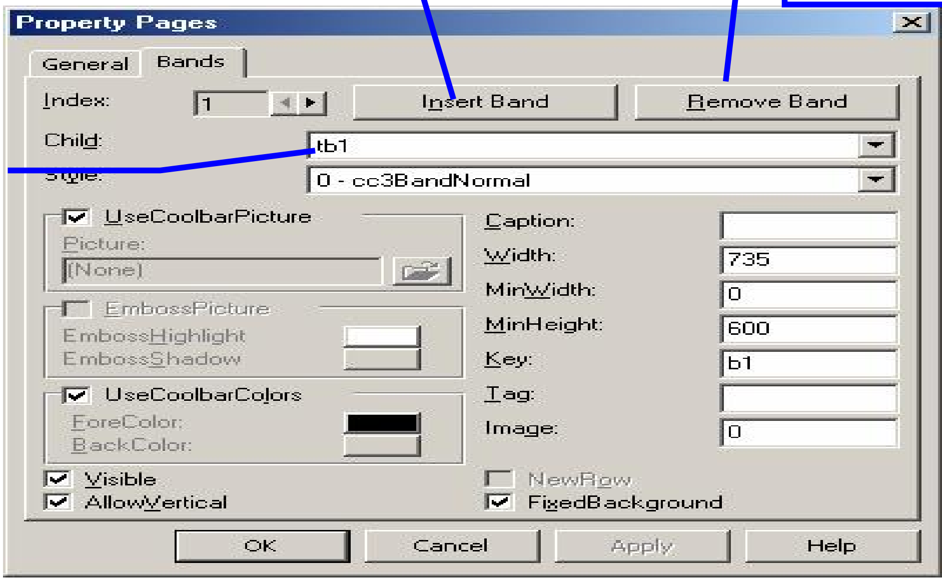
تظهر الشاشة التالية



لإضافة لوح جديد إلى الشريط

لحذف لوح من الشريط

لاختيار شريط الأدوات ويوضع شريط لكل لوح



<u>الطريقة الثانية</u> : باستخدام الكود ويوضع الكود التالي في حدث **form _load**

```
Private Sub Form_Load()
With cb.Bands(1)
    .Key = "b1"
    .Style = cc3BandNormal
End With
With cb.Bands(2)
    .Key = "b2"
    .NewRow = False
    .Style = cc3BandNormal
End With
Set tb1.Container = cb
Set cb.Bands.Item("b1").Child = tb1
cb.Bands(1).MinHeight = tb1.Height
cb.Bands(1).MinWidth = tb1.Width
End Sub
```

جملة **With** تستخدم لتكرار مجموعة من التعليمات على أداة

وضع اسم للوح المراد استخدامه حسب

شكل اللوح اختار عادي

لجعل اللوح في سطر جديد أم لا

لربط شريط الأدوات مع اللوح المطلوب حيث ينفذ هذا الكود مع كل شريط أدوات موجود

يكرر الكود السابق لكل لوح من ألواح الشريط الحاضن

الحلقة الثالثة

بعض الأدوات الهامة وأهم خصائصها

أولا : النماذج : وهي الشاشات التي سيتكون منها البرنامج

كيفية التصميم : لكي اصمم النموذج بصورة جيدة يجب أن اسأل نفسي عدة أسئلة

١ – هل النموذج شاشة رئيسية أم شاشة فرعية

٢ – النموذج سيظهر للمستخدم مكبر أو في الحجم العادي

٣ – هل النموذج يحتوي على أزرار التكبير والتصغير أم سيظهر بدون أزرار

٤ – الألوان المستخدمة في النموذج

٥ – أن النموذج هو أول واجهة المستخدم مما يعطي الانطباع الأول عن البرنامج للمستخدم

أهم خصائص النموذج

اسم الخاصية	معنى الخاصية	شرح الخاصية
Name	اسم النموذج	هو الاسم المميز للنموذج الذي يستخدم داخل الكود
Backcolor	لون الخلفية	اللون الذي يوضع كخلفية للنموذج
Borderstyle	شكل النموذج	حيث يوجد خمسة أشكال للنموذج يظهر بها
Caption	العنوان	هو العنوان الذي يظهر في شريط العنوان للنموذج
Icon	الأيقونة	وهي الرمز الذي يظهر في شريط العنوان جهة اليسار والذي يظهر كرمز للنموذج عند تصغيره
Maxbutton	زر التكبير	الذي منه تستطيع تكبير الشاشة إلى الحد الأقصى أو استعادتها للوضع العادي حيث يمكن تفعيله أو عدم تفعيله ويفضل عدم تفعليه
minbutton	زر التصغير	وهو زر التصغير الذي يصغر الشاشة كأيقونة في شريط المهام حيث يمكن تفعيله أو عدم تفعيله
picture	الصورة	حيث يمكن وضع صورة معبرة كخلفية للنموذج وغالبا نستخدمها للشاشة الرئيسية أو أي شاشة كما نرغب
Rigttolift	من اليمين لليسار	حيث يمكن جعل عنوان النموذج الذي يظهر في شريط العنوان جهة اليمين أو اليسار
Windowsstate	حالة النموذج	من حيث ظهور النموذج عند فتحه هل تريد ظهوره مكبر أو مصغر أو عادي (الوضع الطبيعي للنموذج)

ثانيا : أداة العنوان label 

اسم الخاصية	معنى الخاصية	شرح الخاصية
Name	اسم الأداة	هي الاسم المميز للأداة
alignment	المحاذاة	لمحاذاة النص لليمين أو اليسار أو الوسط
autosize	حجم تلقائي	يجعل حجم الأداة على قدر البيانات المكتوبة بها
Backcolor	لون الخلفية	لوضع لون الخلفية لأداة العنوان
Caption	العنوان	لوضع نص عنوان للأداة
Font	ضبط الخط	لضبط حجم الخط ونوع الخط
forecolor	لون الخط	لضبط لون الخط لأداة العنوان

ثالثا : مربع النص text 

اسم الخاصية	معنى الخاصية	شرح الخاصية
alignment	المحاذاة	لمحاذاة النص لليمين أو اليسار أو الوسط
Backcolor	لون الخلفية	لوضع لون للخلفية
Border style	شكل الإطار	هل تريد مربع النص بارز أم غير بارز
text	نص مربع النص	لوضع نص داخل مربع النص وهي تشبه العنوان
Backcolor	لون الخلفية	لوضع لون الخلفية لأداة العنوان
Font	ضبط الخط	لضبط حجم الخط ونوع الخط
forecolor	لون الخط	لضبط لون الخط لأداة العنوان
locked	تأمين مربع النص	يمكنك تامين المربع من الكتابة فيه ويعرض فقط
Rigttolift	اليمين أو اليسار	لجعل المربع للكتابة العربية أو الإنجليزية
multiline	تعدد السطور	لوضع إمكانية كتابة أكثر من سطر لمربع النص
Passwordchar	حرف كلمة المرور	لجعل مربع النص لإدخال كلمة مرور حيث تضع الحرف الذي سيظهر عند إدخال كلمة المرور
Scrollbars	أشرطة التمرير	حيث يمكن وضع أشرطة تمرير أفقية أو راسية لمربع النص وهي مرتبطة بخاصية تعدد السطر
Tool tip text	تلميحة	حيث يمكنك وضع تلميحة تظهر عند مرور مؤشر الماوس عليها

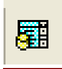
رابعا : زر الأمر command button 

اسم الخاصية	معنى الخاصية	شرح الخاصية
Backcolor	لون الخلفية	لوضع لون للخلفية وهي مرتبطة بخاصية style بمعني يجب تحديد هذه الخاصية أولا حتى يظهر اللون على الزر
cancel	الغاء	حيث يجعل الزر المحدد يفعل عند الضغط على زر ESC في لوحة المفاتيح ودائما يوضع لزر إلغاء الأمر ويوجد زر واحد فقط يأخذ هذه الخاصية
Default	افتراضي	حيث يجعل الزر المحدد يفعل عند الضغط على زر enter في لوحة المفاتيح ودائما يوضع للزر الافتراضي الذي ينفذ عند الضغط على enter ويوجد زر واحد فقط يأخذ هذه الخاصية
Caption	العنوان	لوضع نص عنوان على الزر
font	ضبط الخط	لضبط حجم الخط ونوع الخط
picture	صورة	لوضع صورة كخلفية لزر الأمر وهي مرتبطة بالخاصية style بمعني يجب تحديد هذه الخاصية أولا حتى يظهر اللون على الزر
style	الشكل	لجعل شكل الزر عادي أي لا يستخدم لون الخلفية ولا يتم وضع صورة كخلفية له أو رسومي عكس الأول

خامسا : الأدوات الخاصة بالتعامل مع قاعدة البيانات

مربع قائمة البيانات datalist 

اسم الخاصية	معنى الخاصية	شرح الخاصية
Name	اسم الأداة	الاسم المحدد للأداة والذي يستخدم عند التعامل مع الأداة في الكود
Backcolor	لون الخلفية	
Datafiled	اسم الحقل	تحديد اسم الحقل الذي سيتم عرض بياناته داخل القائمة وهنا تظهر مجموعة من الحقول فخذ أحد الحقول
datasource	مصدر البيانات	وهنا يتم تحديد قاعدة البيانات التي ستغذي الأداة ويتم هنا تحديد أداة قاعدة بيانات معينة
Rowsource	مصدر الصفوف	وهنا يتم تحديد الجدول الذي سيغذي الأداة

مربع سرد البيانات dataCombo 

اسم الخاصية	معنى الخاصية	شرح الخاصية
Name	اسم الأداة	الاسم المحدد للأداة والذي يستخدم عند التعامل مع الأداة في الكود
Backcolor	لون الخلفية	لون الخلفية للأداة
Datafiled	اسم الحقل	تحديد اسم الحقل الذي سيتم عرض بياناته داخل الأداة وهنا تظهر مجموعة من الحقول فخذ أحد الحقول
datasource	مصدر البيانات	وهنا يتم تحديد قاعدة البيانات التي ستغذي الأداة ويتم هنا تحديد أداة قاعدة بيانات معينة
Rowsource	مصدر الصفوف	وهنا يتم تحديد الجدول الذي سيغذي الأداة

وسوف نتعرض لمجموعة أخرى من الأدوات عند شرح بقية الحلقات

ملاحظات هامة على تصميم الأدوات

١ – لا تكثر من استخدام الألوان داخل النموذج

٢ – إذا وجد أداة سوف تستخدمها أكثر من مرة يفضل تنسيق الأداة مرة واحدة ونسخها بالعدد الذي تريده (لاحظ عند اللصق ستظهر رسالة المصفوفة تختار منها لا) .

٣ – إذا كثرت الأدوات داخل النموذج يمكنك جمعها داخل إطار واحد باستخدام أداة الأطارات FRAME

٤ – الأدوات المتشابهة داخل النموذج يكون لها نفس التنسيق مثل العناوين ومربعات النصوص يكون لها نفس التنسيق

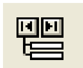
٥ – تحاول دائما التسهيل من عملية إدخال البيانات بالنموذج بمعنى استخدام أسهل الطرق لعملية إدخال البيانات .

الحلقة الرابعة

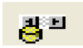
قواعد بيانات فيجوال بيسك

للتعامل مع قواعد البيانات في الفيجوال بيسك يتم ذلك بطريقتين

الطريقة الأولى: عن طريق أدوات قواعد البيانات حيث يوجد مجموعة من الأدوات التي تستخدم للربط بين أدوات البرنامج المنشأ بالفيجوال بيسك وقاعدة البيانات المنشأة بواسطة برنامج قواعد البيانات مثل الأكسس حيث يوجد ثلاث أدوات للتعامل مع قاعدة البيانات

١ – أداة قاعدة البيانات **Data Control** 

٢ – أداة الوصول لقاعدة البيانات عن بعد **Remote Data Control** 

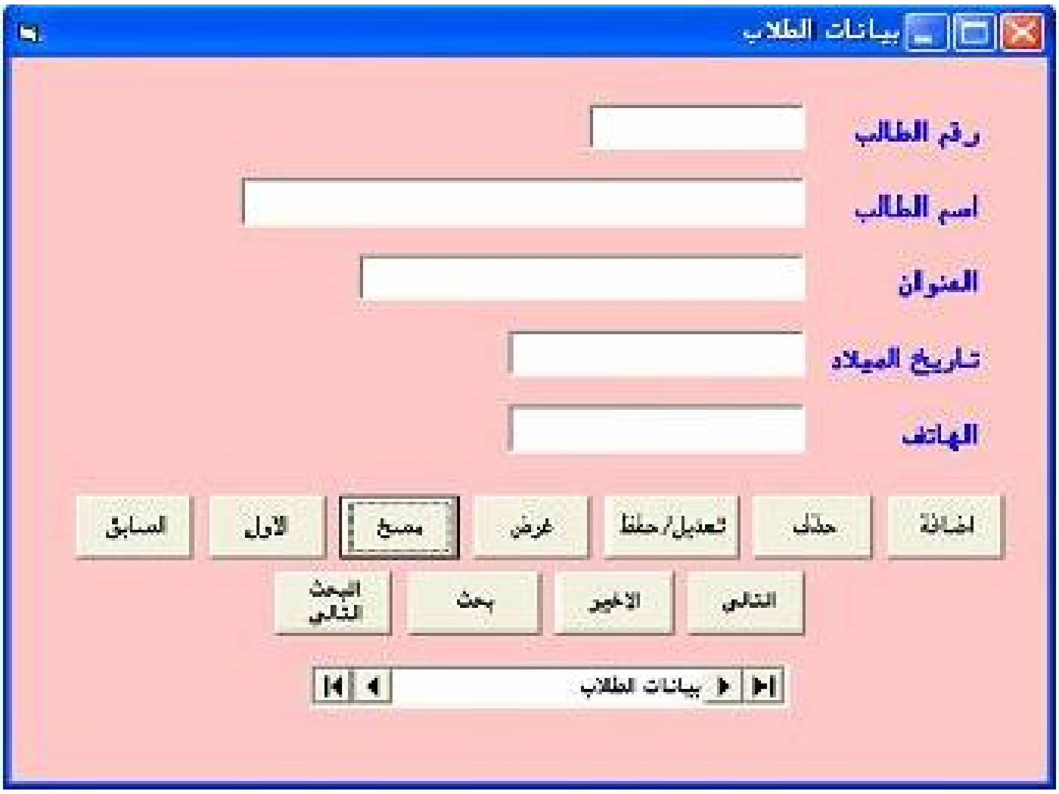
٣ – أداة **Active X Data Control** **(ADODC)** 

وسوف نتعرض بالشرح للأداة الأولى والثالثة

بالمناسبة الأداة الثالثة بها نفس إمكانيات الأداة الثانية وأكثر لذلك سنكتفي بالأداة الأولى والأداة الثالثة

أولا : استخدام أداة قاعدة البيانات Data Control

١ - صمم النموذج التالي



٢ – احفظ المشروع الجديد في مجلد باسم برنامج الطلاب في أي مكان وليكن على القرص الصلب **C**

٣ - صمم قاعدة البيانات بالبرنامج ميكروسوفت أكسس انشأ قاعدة بيانات جديدة باسم **Students** واحفظها في نفس المجلد برنامج الطلاب

٤ – أنشا الجدول التالي باسم **St_inf** وبه الحقول التالية :

اسم الحقل	نوع البيانات
st_no	رقم
st_name	نص
b_date	نص
address	نص
phone	نص

٥ - اضغط على أداة قاعدة البيانات لتحديدها ثم اضبط مجموعة الخصائص التالية :

الخاصية	شرح الخاصية	القيمة
Name	اسم الأداة والذي سوف نستخدمه عندما نشير إلى أداة قاعدة البيانات في الكود	**Data1**
Caption	عنوان الأداة والذي يظهر على الأداة	بيانات الطلاب
connect	اسم البرنامج الذي تريد الاتصال به وهو البرنامج الذي أنشأت به قاعدة البيانات	**Access**
DataBaseName	اسم ومسار قاعدة البيانات	حدد قاعدة البيانات
RecordType	نوع السجلات وسوف نتناول هذه الأنواع بالتفصيل بعد ذلك	**1 - dynast**
RecordSource	اسم الجدول الذي ستستخدمه وهنا سيعرض جميع الجداول الموجودة في قاعدة البيانات التي حددت لها المسار	**St_inf**

وهناك بعض الخصائص سنتعرض لها بالتفصيل بعد ذلك بعد ضبط الخصائص لقاعدة البيانات نأتي للخطوة التالية

٦ – ربط الأدوات مع حقول قاعدة البيانات مثل مربعات النصوص والعناوين والصور حسب الأداة التي تريد ربطها مع الحقل وسنربط هنا مربعات النصوص فقط

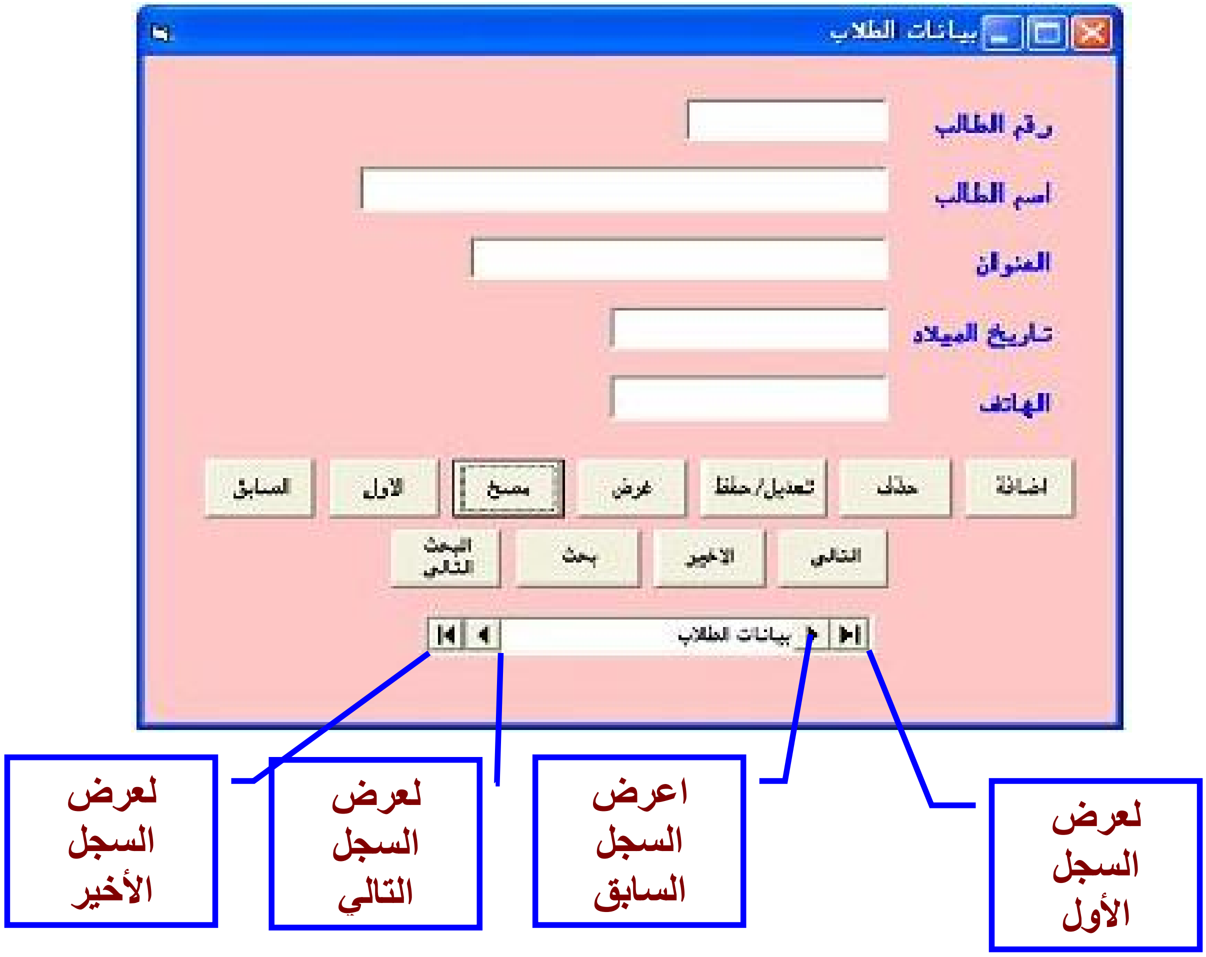
ربط مربعات النصوص

اضغط على مربع النصوص الأول رقم الطالب لتحديده ثم اضبط الخصائص التالية :

الخاصية	شرح الخاصية	القيمة
Name	اسم الأداة والذي سوف نستخدمه عندما نشير إلى مربع النص في الكود	Txt1
DataSource	اسم أداة قاعدة البيانات المطلوب استخدامها	Data1
DataFiled	اسم الحقل المطلوب ربطه مع مربع النص وهنا سيعرض كل الحقول التي في الجدول الذي تم ربطه مع الأداة في الخطوات السابقة	St_no

ملاحظة هامة جدا : وأنا افضل ترك مربعات النصوص بدون ربطها لأن الربط يسبب بعض المشاكل ولكن نفذ الخطوات التالية ثم أزل الأرتباط بعد ذلك عند التعامل مع الكود

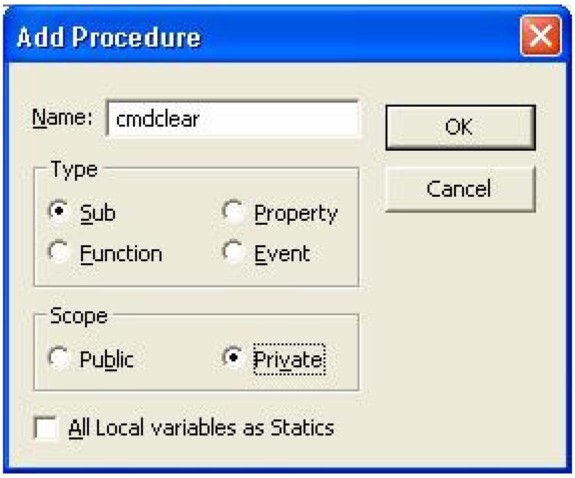
نفذ الخصائص السابقة على كل مربعات النصوص في النموذج فإذا تم فتح النموذج سيظهر أول سجل داخل مربعات النصوص كما يظهر بالصورة التالية



أكواد الأزرار

قبل كتابة الكود يجب أن تجعل خاصية **Data source** و **data Filed** فارغة لكل مربعات النصوص

يمكننا إضافة إجراء لمسح مربعات النصوص افتح قائمة **Tools** واختار **Add procedure**
اكتب اسم الأجراء وليكن **dclear** واختار **Sub** من **Type** و **private** من **scope**



واكتب في الإجراء الكود التالي :

```
Txt1 = ""
Txt2 = ""
Txt3 = ""
Txt4 = ""
Txt5 = ""
Txt1.setfocus
```

أولا : كود زر اضافة : اضغط مرتين على زر اضافة واكتب الكود التالي

```
Data1.Recordset.AddNew
Data1.Recordset![st_no] = Val(Txt1.Text)
Data1.Recordset![st_name] = Txt2.Text
Data1.Recordset![address] = Txt3.Text
Data1.Recordset![b_date] = Txt4.Text
Data1.Recordset![phone] = Txt5.Text
Data1.Recordset.Update
MsgBox "تم حفظ السجل", vbMsgBoxRight + vbMsgBoxRtlReading,"حفظ"
Cmdclear
Data1.Refresh
```

ثانيا : كود زر تعديل : اضغط مرتين على زر تعديل واكتب الكود التالي

```
Data1.Recordset.Edit
Data1.Recordset![st_no] = Val(Txt1.Text)
Data1.Recordset![st_name] = Txt2.Text
Data1.Recordset![address] = Txt3.Text
Data1.Recordset![b_date] = Txt4.Text
Data1.Recordset![phone] = Txt5.Text
Data1.Recordset.Update
MsgBox "تم تعديل السجل", vbMsgBoxRight +
vbMsgBoxRtlReading,"تعديل"
Cmdclear
Data1.Refresh
```

ثالثا : كود زر عرض : اضغط مرتين على زر عرض واكتب الكود التالي

```
Data1.Refresh
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
```

رابعا : كود زر مسح : اضغط مرتين على زر مسح واكتب الكود التالي

```
Cmdclear
```

خامسا : كود زر الأول : اضغط مرتين على زر الأول واكتب الكود التالي

```
Data1.Recordset.MoveFirst
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
```

سادسا : كود زر الأخير : اضغط مرتين على زر الأخير واكتب الكود التالي

```
Data1.Recordset.MoveLast
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
```

سابعا : كود زر التالي : اضغط مرتين على زر التالي واكتب الكود التالي

```
On Error GoTo err:
If Data1.Recordset.EOF Then
MsgBox "هذا آخر سجل"
End If
Data1.Recordset.MoveNext
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
err:
Exit Sub
```

ثامنا : كود زر السابق : اضغط مرتين على زر السابق واكتب الكود التالي

```
On Error GoTo err:
If Data1.Recordset.BOF Then
MsgBox "هذا أول سجل"
End If
Data1.Recordset.MovePrevious
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
err:
Exit Sub
```

تاسعا : كود زر بحث : اضغط مرتين على زر بحث واكتب الكود التالي

سنعرف متغير عام على مستوى النموذج باسم Nam

```
On Error GoTo err:
Dim prompt, title, s, y As String
prompt = "أدخل اسم الطالب"
title = "بحث"
y = InputBox(prompt, title)
If Len(y) = 0 Then
```

```
Exit Sub
Else
nam = y
s = "st_name='" + y + "'"
Data1.Recordset.FindFirst s
If Data1.Recordset.NoMatch = True Then
MsgBox "هذا الطالب غير مسجل"
Else
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
End If
err:
Exit Sub
End If
```

عاشرا : كود زر البحث التالي : اضغط مرتين على زر البحث التالي واكتب الكود التالي

```
Dim s As String
s = "st_name='" + nam + "'"
Data1.Recordset.FindNext s
If Data1.Recordset.NoMatch = True Then
MsgBox "انتهى البحث"
Else
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
End If
```

حادي عشر : كود زر حذف : اضغط مرتين على زر حذف واكتب الكود التالي

```
Dim r As Integer
On Error GoTo err:
r = MsgBox("هل تريد بالتأكيد حذف السجل", vbYesNo, "حذف")
If r = vbYes Then
Data1.Recordset.Delete
Data1.Recordset.MoveNext
```

```
Txt1.Text = Data1.Recordset![st_no]
Txt2.Text = Data1.Recordset![st_name]
Txt3.Text = Data1.Recordset![address]
Txt4.Text = Data1.Recordset![b_date]
Txt5.Text = Data1.Recordset![phone]
MsgBox ("تم حذف السجل")
Data1.Refresh
Else
Exit Sub
End If
err:
Exit Sub
```

الحلقة الخامسة

كائنات الوصول للبيانات Data Access Object (DAO)

تحدثنا أن هناك ثلاث أدوات تستخدم لربط قاعدة البيانات المنشأة بأحد برامج قواعد البيانات مثل برنامج ميكروسوفت أكسس وقلنا أن هناك ثلاث أدوات ولكل أداة من هذه الأدوات يمكن استخدام كائن من الكائنات بدلا من الأداة الرسوميه وهي موجودة بالجدول التالي

م	اسم الأداة	الكائن المقابل
١	١ – أداة قاعدة البيانات Data Control	كائن الوصول للبيانات Data Access Object (DAO)
٢	٢ – أداة الوصول لقاعدة البيانات عن بعد Remote Data Control	كائن الوصول للبيانات عن بعد Remote Access Object (RDO)
٣	٣ – أداة Active X Data Control (ADODC)	كائن اكتيف اكس للوصول للبيانات ActiveX Access Object (ADO)

ولقد قمنا بشرح الأداة Data Control في الحلقة السابقة حيث يمكنك استخدام أكثر من أداة واحدة داخل النموذج الواحد للربط مع أكثر من جدول وسنقوم في هذه الحلقة بشرح الكائن المقابل لهذه الأداة وهو كائن الوصول للبيانات DAO .

الكائن Object هو صنف له مجموعة من الخصائص والطرق والأحداث لتأدية غرض معين وكائن الوصول لقاعدة البيانات DAO هو الرابط بين برنامج الفيجوال بيسك وبرنامج قواعد البيانات حيث يشبه الوصول للبيانات بالشكل التالي :

برنامج العميل VB ← الكائن DAO أو RDO ← كائن الاتصال ODBC ← برامج Data Base

حيث يمثل (ODBC) Open Database Connectivity عنصر الاتصال مع قاعدة البيانات والذي يسمى الاتصال المفتوح مع قاعدة البيانات وهو بداخلة مجموعة من الكائنات يقع في أعلاها الكائن DAO و RDO اللذان يستخدمان المحرك ODBC للوصول للبيانات وهناك مجموعة أخرى من الكائنات سوف نتعرض لها عند الشرح .

وأدخل معي الى عالم البرمجة الحقيقة حيث كل شيء ميسر هنا ولكن يبقي الابتكار

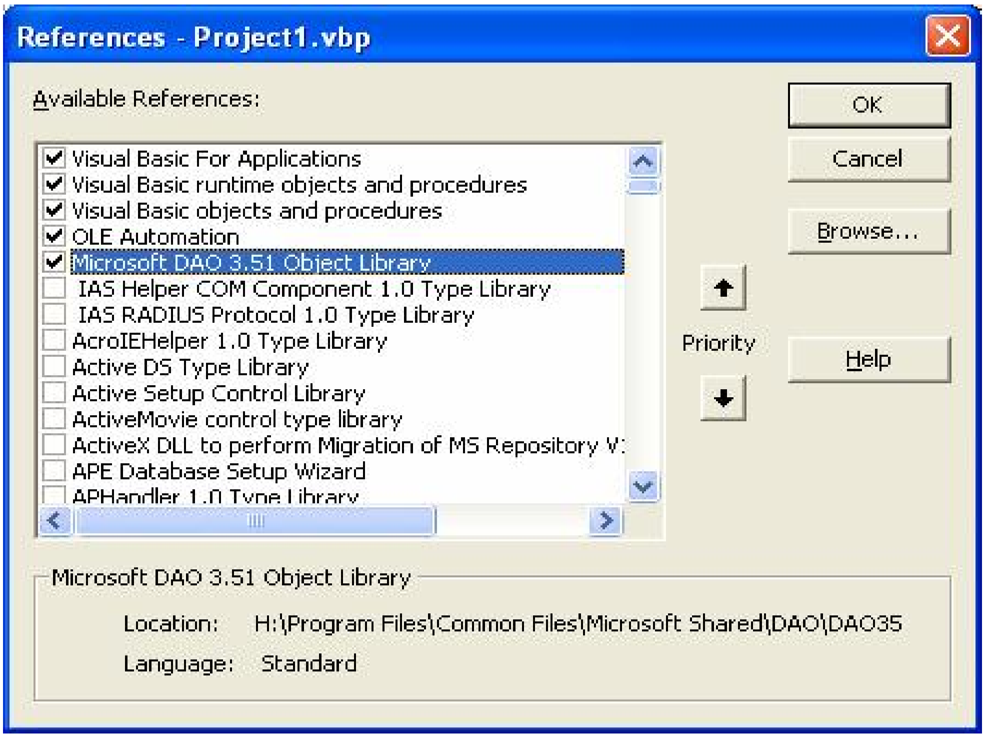
خطوات التعامل مع الكائنات DAO

الخطوة الأولى

تعريف الكائنات في مرجع البرنامج References وهنا يجب التفريق بين Component وهو المكان الذي تضيف منه أدوات جديدة أما References هو المكان الذي تضيف منه كائنات جديدة ولإضافة الكائنات نقوم بالخطوات التالية

١ - اضغط قائمة PROJECT واختار REFRANCES

٢ – من الشاشة التي تظهر أختار الكائن MICROSOFT DAO 3.51 OBJECT LIBRARY أو حسب آخر إصدار موجود عندك



الخطوة الثانية

إنشاء قاعدة البيانات حيث يمكنك إنشاء قاعدة البيانات والجداول باستخدام الكود أو باستخدام برنامج مخصص لقاعدة البيانات (المفضل)

الخطوة الثالثة

في شاشة محرر الكود للنموذج وفي منطقة تعريف المتغيرات للنموذج (أعلى النموذج) يتم تعريف المتغيرات التي تعبر عن كائنات قاعدة البيانات كالتالي :

Dim WS As DAO.Workspace تعريف متغير يعبر عن منطقة العمل

Dim DB As DAO.Database تعريف متغير يعبر عن كائن قاعدة البيانات

Dim RS As DAO.Recordset تعريف متغير يعبر عن كائن مجموعة السجلات

طبعا معروف أن المتغير إذا كنت ستستخدمه داخل المشروع بالكامل فيفضل إنشاء ملف برمجة Module وتضع فيه المتغيرات وهذا ما أفضله أن تضع منطقة العمل وقاعدة البيانات في ملف برمجة

الخطوة الرابعة

وهنا يجب أن نعرف أنه في قواعد البيانات يفضل أن تكتب كود فتح قاعدة البيانات مرة واحدة في حدث Form load للشاشة الرئيسية للبرنامج وهي الشاشة التي تعتبر الحاوية لكل شاشات البرنامج ولا تغلق إلا عند الخروج من البرنامج أما الجداول فيتم فتحها وإغلاقها حسب الاستخدام .

إنشاء قاعدة البيانات والجداول باستخدام الكود

إذا كنت ترغب في إنشاء قاعدة البيانات وإنشاء الجداول باستخدام الكود فيتم ١ –

١ - تعريف الكائنات التالية في نفس الموضع السابق

Dim TBL1 As TableDef تعريف متغير يعبر عن كائن الجداول (الجدول الأول)

Dim TBL2 As TableDef تعريف متغير يعبر عن كائن الجداول (الجدول الثاني)

Dim f1, f2, f3, f4, f5, f6 As Field تعريف متغير يعبر عن حقول الجدول الأول

Dim ff1, ff2, ff3, ff4, ff5, ff6 As Field تعريف متغير يعبر عن حقول الجدول الثاني

٢ – اضغط على زر إنشاء قاعدة البيانات ثم أكتب الكود التالي :

```
Dim a, r
Dim s As String
s = App.Path & "\" & "students.mdb"    '
Set ws = DBEngine.Workspaces(0)
a = Dir(s)    'للبحث عن ملف قاعدة البيانات في مسار معين
If a = "" Then   'الملف غير موجود
 Set db = ws.CreateDatabase(s, dbLangGeneral)  'إنشاء قاعدة البيانات
Else
```

```
r = MsgBox("قاعدة البيانات موجودة هل تريد استبدالها", vbYesNo, "استبدال")
 If r = vbYes Then
 Kill (s)   ' حذف قاعدة البيانات
 Set db = ws.CreateDatabase(s, dbLangGeneral)   'أعادة الإنشاء
 Else
 Exit Sub
 End If
End If
```

<u>ملاحظات على الكود السابق</u>

١ – جملة App.Path ترجع بمسار الملف وفي هذه الحالة يفضل وضع ملف قاعدة البيانات في نفس مجلد البرنامج

٢ – جملة DBEngine وهو محرك قاعدة البيانات الخاص بقادة البيانات اكسس

٣ – جملة CreateDatabase تنشأ قاعدة البيانات الجديدة والجملة تكتب على الشكل

Set database = workspace.CreateDatabase (name, locale, options)

Name : اسم ومسار قاعدة البيانات

Local : لغة قاعدة البيانات

Option : مجموعة من الإختيارات مثل التشفير وأصدارقاعدة البيانات ....

٤ – جملة Dir(path) للبحث عن ملف لمسار معين

٥ – حملة kill(path) لحذف ملف من خلال مسار معين .

<u>إنشاء الجداول باستخدام الكود</u>

يتم إنشاء الجداول باستخدام الكود حيث نحتاج في بعض الأحيان إلى إنشاء جدول إثناء التنفيذ فيتم كما يلي .... اضغط على الزر إنشاء الجداول

```
Dim s As String
s = App.Path & "\" & "students.mdb"
Set ws = DBEngine.Workspaces(0)
Set db = ws1.OpenDatabase(s)
Set tbl1 = db.CreateTableDef("st_inf")
Set f1 = tbl1.CreateField("st_no", dbInteger)
Set f2 = tbl1.CreateField("st_name", dbText, 40)
Set f3 = tbl1.CreateField("address", dbText, 40)
Set f4 = tbl1.CreateField("phone", dbText, 40)
Set f5 = tbl1.CreateField("b_date", dbDate)
```

```
tbl1.Fields.Append f1
tbl1.Fields.Append f2
tbl1.Fields.Append f3
tbl1.Fields.Append f4
tbl1.Fields.Append f5
db1.TableDefs.Append tbl1
```

حيث يتم فتح قاعدة البيانات التي تم إنشائها ثم إنشاء الجداول وإنشاء الحقول بإضافة اسم الحقل ونوع البيانات وحجم الحقل ثم إضافة الحقول إلى الجدول ولإضافة الجدول إلى قاعدة البيانات بعد ذلك .

التعامل مع قاعدة البيانات التي تم إنشائها

١ - في الحدث Form Load للنموذج يتم فتح قاعدة البيانات والجداول المراد استخدامها داخل النموذج

ويفضل إنشاء ملف برمجة حيث تعرف فيه متغير قاعدة البيانات ومتغير مجموعة العمل حتى يمكنك استخدامها داخل البرنامج بالكامل دون تعريفها في كل نموذج ويتم فتح قاعدة البيانات في حدث التحميل للشاشة الرئيسية للبرنامج كالتالي :

اضغط على مكان خالي بالنموذج مرتين ثم اكتب الكود التالي

Dim s as string

S = app.path & "\students.mdb" **هذا المتغير يعبر عن مسار قاعدة البيانات المراد فتحها**

Set db = OpenDatabase(S) **هذه الجملة لفتح قاعدة البيانات**

٢ - بعد ذلك افتح الجدول المطلوب في كل شاشة من شاشات البرنامج يتم تعريف متغير للتعبير عن الجدول عند التعامل معه مثل RS حيث يوضع التعريف التالي في منطقة تعريف المتغيرات في النموذج

Dim RS as DAO.Recorset

ثم بعد ذلك يتم فتح الجدول في أي مكان نريد استخدامه

Set rs = db.OpenRecordset("st_info", dbOpenDynaset)

حيث أن جملة OpenRecordset لها بارامترين

الأول : يحدد اسم الجدول المطلوب فتحة

الثاني نوع السجلات التي ستستخدمها

أنواع السجلات التي يمكن فتحها بالكائن DAO

النوع الأول Table Recordset

وهذا النوع هو ارتباط مباشر لأي جدول في قاعدة البيانات

مميزاته

١ – استخدام نظام الفهرسة INDEX

٢ – إتاحة عمليات بحث سريعة

عيوبه

١ – عدم إمكانية إنشاء تصفية للسجلات

٢ – استخدام تعليمة SEEK في البحث وهى تبحث عن السجل الأول فقط

٣ – عدم استخدام تعليمة البحث FIND التي يمكن أن توجد عدد كبير من السجلات

ولفتح جدول باستخدام هذا النوع نستخدم الجملة التالية

```
Set rs = db.OpenRecordset("st_info", dbOpenTable)
```

النوع الثاني DynaSet Recordset

وهو عبارة عن مجموعة من السجلات من جدول واحد أو أكثر في قاعدة البيانات

مميزاته

١ – إمكانية ربط البيانات في أكثر من جدول

٢ – إمكانية استخدام الأمر FIND في البحث عن السجلات

٣ – إمكانية عمل تصفية للسجلات

عيوبه

١ - لا يستخدم الفهارس INDEX

٢ – لا يتيح تعديل السجلات أو حذفها بسهولة

لفتح جدول باستخدام هذا النوع من السجلات نستخدم الجملة التالية

```
Set rs = db.OpenRecordset("st_info", dbOpenDynaset)
```

النوع الثالث : SnapShot Recordset

وهى عبارة عن صورة من الجدول كما هو وهي للقراءة فقط

مميزاته

١ - إمكانية ربط أكثر من جدول

٢ - إمكانية استخدام أمر البحث FIND

٣ - سرعة البحث للسجلات لأنها في الذاكرة

عيوبه

١ – عدم إمكانية تحديث وتعديل البيانات لأنها للقراءة فقط

٢ – عدم إمكانية استخدام الفهارس

٣ – استهلاك الذاكرة وخاصة عند فتح أكثر من جدول

لفتح جدول باستخدام هذا النوع من السجلات نستخدم الجملة التالية

```
Set rs = db.OpenRecordset("st_info", dbOpenSnapShot)
```

الحلقة السادسة

العمليات الأساسية على السجلات للكائنات DAO

وهي العمليات الأساسية على قاعدة البيانات باستخدام الكائن DAO

نراجع سريعا الخطوات السابقة

١ – تصميم النموذج ووضع مربعات النصوص اللازمة .

٢ – تعريف كائن قاعدة البيانات في مرجع البرنامج MICROSOFT DAO 3.51

٣ – تعريف متغيرات تعبر عن كائنات قاعدة البيانات وهنا يفضل إنشاء ملف برمجة لتعريف متغير قاعدة البيانات وتعريف متغيرات تعبر عن الجداول حيث تضع متغير لكل جدول تريد استخدامه

٤ – تقوم بفتح قاعدة البيانات في حدث Form_load للشاشة الرئيسية للبرنامج

وهيا بنا إلى العمليات الرئيسية على قاعدة البيانات

كود إضافة سجل جديد

١ - لإضافة سجل جديد إلي قاعدة البيانات نكتب الكود التالي اضغط على زر إضافة مرتين يجب أولا فتح الجدول المطوب التعامل معه باستخدام الجملة

```
Set rs = db1.OpenRecordset("st_inf", dbOpenDynaset)
```

٢ – التأكد أن مربعات النصوص غير خالية باستخدام الكود التالي

```
If Txt1.Text = "" or Txt2.Text = "" or Txt3.Text = "" or Txt4.Text <>
"" or Txt5.text = "" Then
    MsgBox "يجب إدخال القيم في مربعات النص"
Exit Sub
End If
```

٣ – التأكد من أن الجدول به سجلات أم لا باستخدام الكود التالي بحيث انه إذا كان به سجلات يتم البحث عن السجل الحالي موجود أو غير موجود حتى نضمن عدم تكرار السجلات داخل الجدول فإذا كان الجدول خالي من السجلات يتم الانتقال لجملة لتعبئة السجل الجديد مباشرة

```
If  rs.Recorcount = 0 then go to ss
```

٢ – إذا كان يوجد سجلات داخل الجدول يتم التأكد هل رقم الطالب الحالي (السجل الحالي) موجود أم لا فإذا كان موجودا يتم تعديله أو أي إجراء آخر وإلا يذهب إلى العنوان ss لإضافة سجل جديد

```
rs.MoveFirst
Do While Not rs.EOF
If Not rs.EOF Then
     If rs![st_no] = Val(Txt1) Then
     r = MsgBox ("السجل موجود من قبل هل تريد التعديل",vbYesNo,"تعديل")
      If r = vbYes Then
        rs.Edit
        rs![st_no] = Val(Txt1)
        rs![st_name] = Txt2
        rs![address] = Txt3
        rs![b_date] = Txt4
        rs![phone] = Txt5
        rs.Update
        MsgBox "تم تعديل السجل"
        cmdclear_Click 'استدعاء حدث النقر على زر المسح
    Else
     Txt1.SetFocus
     Exit Sub
   End If
  Exit Sub
 End If
End If
rs.MoveNext
Loop
```

تعديل السجلات

لتعديل سجل معين نستخدم الكود التالي اضغط على زر تعديل مرتين

```
rs.Edit
rs![st_no] = Val(Txt1)
rs![st_name] = Txt2
rs![address] = Txt3
```

```
rs![b_date] = Txt4
rs![phone] = Txt5
rs.Update
```

الكود الكامل لحفظ وتعديل السجلات (زر حفظ /تعديل)

هذا الكود للإضافة والتعديل معا وهذا الكود يمكن استخدامه بصورة عامة في أي زر إضافة

```
Dim r As String
If Txt1.Text = "" Or Txt2.Text = "" Or Txt3.Text = "" Or Txt4.Text =
"" Or Txt5.Text = "" Then
MsgBox("من فضلك اكمل البيانات")
Txt1.SetFocus
Exit Sub
End If
If rs.RecordCount = 0 Then GoTo Ss
rs.MoveFirst
Do While Not rs.EOF
If Not rs.EOF Then
If rs![st_no] = Val(Txt1) Then
r = MsgBox ("السجل موجود من قبل هل تريد التعديل",vbYesNo"تعديل")
If r = vbYes Then
rs.Edit
rs![st_no] = Val(Txt1)
rs![st_name] = Txt2
rs![address] = Txt3
rs![b_date] = Txt4
rs![phone] = Txt5
rs.Update
MsgBox" تم تعديل السجل"
cmdclear_Click   'استدعاء حدث النقر على زر المسح
Else
Txt1.SetFocus
Exit Sub
End If
Exit Sub
End If
End If
rs.MoveNext
```

```
Loop
Ss:
rs.AddNew
rs![st_no] = Val(Txt1)
rs![st_name] = Txt2
rs![address] = Txt3
rs![b_date] = Txt4
rs![phone] = Txt5
rs.Update
cmdclear_Click    'استدعاء حدث النقر على زر المسح حذف سجل محدد
```

كود حذف سجل

لحذف سجل محدد يكتب الكود التالي

```
Dim r As Integer
 r = MsgBox ("هل تريد حذف السجل الحالي" ,vbYesNo,"الحذف" )
     If rs.RecordCount = 0 Then
     MsgBox" لا يوجد سجلات داخل الجدول"
     Exit Sub
     End If
   If r = vbYes Then
      Do While Not rs.EOF
      If rs![st_no] = Val(Txt1) Then
       rs.Delete
       rs.Fields.Refresh
       If rs.RecordCount = 0 Then
       MsgBox" لا يوجد سجلات داخل الجدول"
       Exit Sub
       End If
       rs.MoveFirst
      Txt1.Text = rs![st_no]
      Txt2.Text = rs![st_name]
      Txt3.Text = rs![address]
      Txt4.Text = rs![b_date]
      Txt5.Text = rs![phone]
      Exit Sub
       End If
       rs.MoveNext
```

```
     Loop
  Else
   Txt1.SetFocus
   Exit Sub
 End If
```

كود البحث باستخدام جملة FindFirst

البحث عن سجل معين باستخدام جملة FINDfirst

للبحث عن سجل باستخدام جملة FIND يتم كتابة الكود التالي :

يتم تعريف متغير نصي في أعلى النموذج منطقة تعريف المتغيرات كالتالي

```
Dim name as string     'لاستخدام القيمة التي تخزن فيه للبحث عن السجل التالي
```

ثم يتم كتابة الكود التالي

```
Dim prompt As String
Dim title As String
Dim s As String
Dim y As String
Prompt = "ادخل اسم الطالب"
Title = "البحث"
y = InputBox(prompt, title)    القيمة التي يدخلها المستخدم تخزن في المتغير
 If Len(y) = 0 Then   إذا كان طول المتغير يساوي صفر بمعنى أن المستخدم لم يدخل
 اسم للبحث
  Exit Sub
 Else
  name = y
  s = "st_name=' " + y + " ' "
  rs.FindFirst s           جملة البحث عن السجل الأول الذي يمثل الاسم الذي
أدخله المستخدم
    If rs.NoMatch Then              تستخدم هذه الجملة للتحقق من العثور
على الاسم أم لا
     MsgBox"الاسم غير موجود "
    Else
     Txt1 = rs![st_no]
     Txt2 = rs![st_name]
```

```
      Txt3 = rs![address]
      Txt4 = rs![b_date]
      Txt5 = rs![phone]
    End If
 End If
```

كود البحث عن السجل التالي باستخدام FindNext

البحث عن السجل التالي باستخدام جملة FIND

```
Dim s As String
s = "st_name=' " + nam + " ' "
 rs.FindNext s
 If rs.NoMatch Then
   MsgBox "انتهى البحث"
 Else
     Txt1.Text = rs![st_no]
     Txt2.Text = rs![st_name]
     Txt3.Text = rs![address]
     Txt4.Text = rs![b_date]
     Txt5.Text = rs![phone]
 End If
```

البحث باستخدام جملة DO WHILE

للبحث عن سجل باستخدام جملة DO WHILE نكتب الكود التالي :

```
Dim a As String
a = InputBox("ادخل اسم الطالب","بحث")
rs.MoveFirst
Do While Not rs.EOF      'كرر لنهاية الجدول
If rs![st_name] = a Then
Text1 = rs![St_no]
Text2 = rs![st_name]
Text3 = rs![ address]
Text4 = rs![b_date]
Text5 = rs![phone]
Exit Sub
End If
rs.MoveNext
```

```
Loop
```

التحرك إلى السجل الأول بجملة Movefirst

للتحرك إلى السجل الأول في الجدول وعرضه نكتب الكود التالي :

```
If rs.RecordCount = 0 Then
  MsgBox"لا يوجد سجلات داخل الجدول"
  Exit Sub
End If
rs.MoveFirst
Text1 = rs![St_no]
Text2 = rs![st_name]
Text3 = rs![ address]
Text4 = rs![b_date]
Text5 = rs![phone]
```

التحرك للسجل الأخيرباستخدام جملة MoveLast

للتحرك للسجل الأخير من الجدول وعرضه نكتب الكود التالي :

```
If rs.RecordCount = 0 Then
  MsgBox"لا يوجد سجلات داخل الجدول"
  Exit Sub
  End If
rs.MoveLast
Text1 = rs![St_no]
Text2 = rs![st_name]
Text3 = rs![ address]
Text4 = rs![b_date]
Text5 = rs![phone]
```

التحرك للسجل التالي باستخدام جملة MoveNext

للتحرك للسجل التالي من الجدول وعرضه نكتب الكود التالي :

```
If rs.RecordCount = 0 Then
```

```
  MsgBox"لا يوجد سجلات داخل الجدول"
  Exit Sub
End If
rs.MoveNext
 If rs.EOF Then
    MsgBox"هذا هو آخر سجل"
 Exit Sub
 End If
Text1 = rs![St_no]
Text2 = rs![st_name]
Text3 = rs![ address]
Text4 = rs![b_date]
Text5 = rs![phone]
```

التحرك للسجل السابق باستخدام جملة MovePrevious

للتحرك للسجل السابق من الجدول وعرضه نكتب الكود التالي :

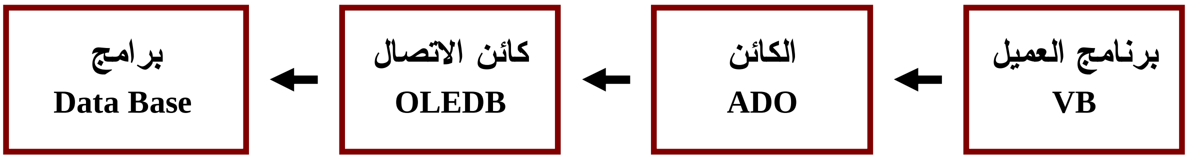
```
If rs.RecordCount = 0 Then
  MsgBox"لا يوجد سجلات داخل الجدول"
  Exit Sub
End If
rs.MovePrevious
  If rs.BOF Then
     MsgBox"هذا هو أول سجل"
  Exit Sub
  End If
Text1 = rs![St_no]
Text2 = rs![st_name]
Text3 = rs![address]
Text4 = rs![b_date]
Text5 = rs![phone]
```

الحلقة السابعة

التعامل مع الكائن ActiveX Data Object (ADO)

حيث حان الوقت لندخل لعالم البرمجة الحقيقية حقيقية حقيقية أخواني ستجدوا مع هذا الكائن الرائع المتعة الكاملة وحان الآن أيضا للتحول من الكائنات القديمة DAO,RDO إلى هذا الكائن الذي به جميع الإمكانيات التي بالكائنين السابقين وأكثر ٠

وسبب التأخر في هذه الحلقة أني كنت سأكتب موضوعين قبل هذا الموضوع كتهيئة لهذا الكائن فكنت سأتناول مجموعة من الأدوات التي تتعامل مع قاعدة البيانات مثل Data list و data Combo وdata Grid و Flex grid وكذلك كنت سأخصص حلقة عن جمل SQL داخل الفيجوال ولكن قلت نبدأ بالكائن ثم نخصص هاتين الحلقتين للكائن ADO فهيا بنا الى هذا الكائن .



خطوات التعامل مع الكائنات ADO

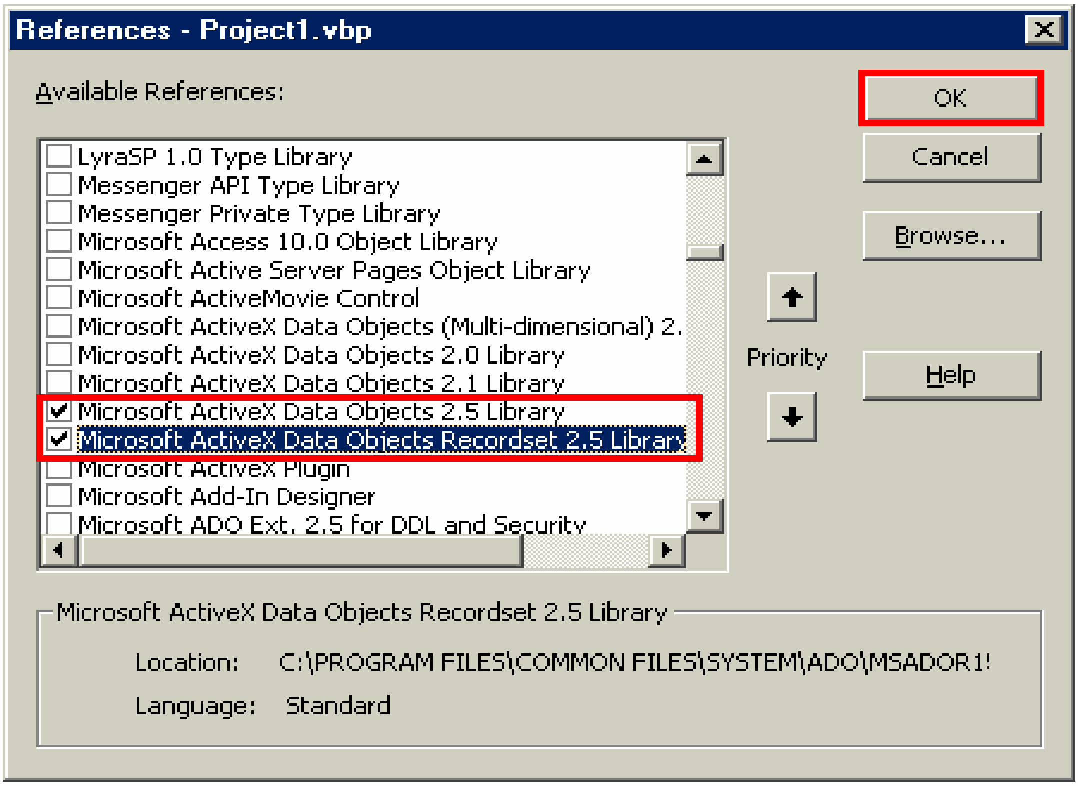
الخطوة الأولى :

تعريف الكائنات ADO مثلما تم مع الكائن DAO في مرجع البرنامج References كالتالي :

١ - اضغط قائمة Project واختار References

٢ – من الشاشة التي تظهر أختار الكائنين

MICROSOFT ACTIVE X DATA OBJECT 2.5 LIBRARY
MICROSOFT ACTIVE X DATA OBJECT RECORDSET 2.5 LIBRARY



<u>الخطوة الثانية :</u>

إنشاء قاعدة البيانات عن طريق برنامج MICROSOFT ACCESS أو برنامج SQL Server أو برنامج Oracle وسوف نتناول كيفية التعامل مع الأنواع الثلاث وهذه هي الميزة الرائعة للكائن ADO والذي يمكنك من التعامل مع عدد ضحم من برامج قواعد البيانات الضخمة .

<u>الخطوة الثالثة :</u>

تصميم النموذج بوضع مربعات نصوص تعبر عن حقول الجدول المراد استخدامه ووضع أزرار الأوامر المراد استخدامها في النموذج مثل (إضافة – بحث ....) كما صنعنا في شاشة بيانات الطلاب .

<u>الخطوة الرابعة</u>

تعريف متغيرات تعبر عن قاعدة البيانات والجدول التي ستستخدمها ويفضل أنشاء ملف برمجة Module لنعرف فيه كائن قاعدة البيانات كالتالي :

```
public db as New ADODB.Connection
```

وكذلك يتم تعريف متغيرات تعبر عن الأوامر التي يمكن تنفيذها على قاعدة البيانات وكذلك متغير يعبر عن مجموعة السجلات من جدول أو أكثر والمتغيرات من هذا النوع تعرف داخل كل نموذج على حدة فمثلا في أحد الشاشات سنستخدم الجدولين مثلا جدول العملاء وجدول الأصناف فيتم تعريف متغيرين كالتالي

```
Dim rscust As New ADODB.Recordset
dim rsitem as New ADODB.Recordset
```

وهكذا مع كل جدول سنستخدمه .

الخطوة الخامسة

في الحدث **Form_Load** للنموذج الرئيسي للبرنامج يتم فتح قاعدة البيانات ويفضل أن يتم فتح قاعدة البيانات في الشاشة الرئيسية للبرنامج وتظل مفتوحة دائما طالما البرنامج يعمل وهنا نستخدم الأكواد التالية حسب قاعدة البيانات المستخدمة

أولا : كود فتح قاعدة البيانات أكسس

```
Dim S As String
S = "provider =microsoft.jet.OLEDB.4.0;"
S = s & "data source=" & App.Path & "\students.mdb"
db.ConnectionString = s
db.Open
```

ملاحظات على الكود السابق

١ – المزود **Provider** وهو اسم السائق الخاص بكائن الاتصال بقاعدة البيانات **OLEDB** والذي سيستخدمه الكائن **ADO** للوصول لقاعدة البيانات وهو هنا مع برنامج ميكروسوفت أكسس يكون **microsoft.jet.OLEDB.4.0**

٢ – **Data Source** يوضع فيه مسار قاعدة البيانات حيث يمكنك وضع المسار الكامل لقاعدة البيانات مثل **"c:\data base\students.mdb"** أو استخدام الدالة **App.path** إذا كانت قاعدة البيانات مخزنة في نفس مجلد البرنامج .

٣ – **Connectionstring** هي الجملة التي تزود لقاعدة البيانات المزود المستخدم ومسار واسم قاعدة البيانات ويمكنك كتابة الجملة مباشرة كالتالي

```
db.connectionstring = "provider = microsoft.jet.OLEDB.4.0;" & "data source=" & App.Path & "\students.mdb"
```

٤ – ثم بعد ذلك يتم فتح قاعدة البيانات باستخدام الطريقة Open

ثانيا : كود فتح قاعدة البيانات SQL Server

```
Dim S As String
S = "provider = SQLOLEDB.1;user ID = sa ;password = ;"
S = s & "Initial Catalog =students;" & data source =nasser"
db.ConnectionString = s
db.Open
```

ملاحظات على الكود السابق

١ – المزود Provider وهو اسم السائق الخاص بكائن الاتصال بقاعدة البيانات OLEDB والذي سيستخدمه الكائن ADO للوصول لقاعدة البيانات وهو هنا مع برنامج SQL server يكون SQLOLEDB.1 .

٢ – رقم المستخدم User ID وهو رقم تعريف المستخدم لقاعدة البيانات حيث رقم المستخدم الافتراضي هو sa حيث من خلاله يمكنك وضع الحقوق والصلاحيات للوصول لقاعدة البيانات لكل مستخدم .

٣ – كلمة المرور Password وهي تستخدم للتحقق من المستخدم وإذا كنت تود استخدام الأمن المتكامل في ويندوز ٢٠٠٠ أو ويندوز NT فنستخدم بدلا من رقم المستخدم وكلمة المرور user ID = sa ;password = نستخدم الجملة Integrated Security = SSPI

٤– Data Source يوضع فيه اسم خادم SQL Server الذي تستخدمه فعند التعامل مع SQL Server حتى تستطيع إنشاء قاعدة بيانات يجب أن تعرف خادم قاعدة البيانات ويكون له اسم وغالبا يأخذ نفس اسم الجهاز وليكن هنا Nasser فإذا كنت تنشأ برنامج لشبكة شركة يجب أن تعرف من مدير الشبكة اسم خادم قاعدة البيانات .

٥ - Initial Catalog وهي الميزة الرائعة لقاعدة البيانات SQL Server أن قاعدة البيانات لا تخزن في ملفات منفصلة بل تخزن على الخادم الرئيسي والذي يمكن الوصول إليه من أي جهاز عميل وهنا تضع اسم قاعدة البيانات المنشأة على الخادم Students ويوضع بدون مسار .

٦ – Connectionstring هي الجملة التي تزود لقاعدة البيانات المزود المستخدم .

٧ – ثم بعد ذلك يتم فتح قاعدة البيانات باستخدام الطريقة Open .

الخطوة السادسة

فتح الجدول المطلوب وكما قلنا سابقا انه يتم تعريف متغير لكل جدول ستستخدمه ويوضع التعريف في منطقة تعريف المتغيرات في أعلى النموذج ثم يتم فتح الجدول المطلوب بكتابة الكود التالي :

```
If rsst.State = adStateOpen Then rsst.Close
rsst.Open "st_inf", db, adOpenDynamic, adLockOptimistic
```

ملاحظات على الكود السابق

١ - حيث السطر الأول يتأكد أن الجدول غير مفتوح فإذا كان مفتوح يتم إغلاقه حتى لا يحدث خطأ أنك تقوم بفتح عنصر مفتوح قبل ذلك .

٢ - والسطر الثاني جملة فتح جدول وهي تحتوي على أربع بارامترات

الأول: اسم الجدول المراد فتحة ويوضع بين علامتي تنصيص وإن كان الاسم يحتوي على كلمتين منفصلتين ينبغي وضعه بين قوسين"[st Info]" .

الثاني : اسم قاعدة البيانات التي تستخدمها وهنا نضع المتغير الذي استخدمناه .

الثالث : نوع المؤشر courser type أو نوع السجلات وهنا يوجد أربعة انواع من المؤشرات أو السجلات

ملاحظة : يطلق على مجموعة الصفوف العائدة من الجدول اسم السجلات عند استخدام أكسس أو المؤشرات عند استخدام SQL Server .

النوع الأول DYNAMIC RECORDSET

وهذا النوع من أكثر الأنواع مرونة حيث يتم التعامل مع السجلات عن طريق مؤشرات تعبر عن السجلات الحقيقة في الجدول ولذلك تستهلك مقدار من الذاكرة

ويتم استخدام هذا النوع كالتالي

```
rs.Open " st_inf ", db, adOpenDynamic, adLockOptimistic
```

النوع الثاني FORWARDONLY RECORDSET

وهي تسمح بالتحرك داخل السجلات في الاتجاه للأمام فقط حيث يمكن استخدام جملة MoveNext ولا يمكن استخدام جملة MovePrevious ويتم فتح الجدول بهذا النوع من السجلات كالتالي

rs.Open " st_inf ", db, adOpenForeardonly, adLockOptimistic

النوع الثالث : KEYSET RECORDSET

يشبه نوع السجلات Static ولكن يتميز عنه قدرته على رؤية التغييرات التي تتم على السجلات ويمكن التحرك داخل السجلات للمام وللخلف يتم فتح هذا النوع من السجلات كالتالي

rs.Open " st_inf ", db, adOpenKeyset, adLockOptimistic

النوع الرابع : STATIC RECORDSET

وهو يشبه إلى حد بعيد Snapshot في كائنات DAO وهو للقراءة فقط ولا يرى التعديلات التي تقع على السجلات ويمكن فتح هذا النوع من السجلات كالتالي

rs.Open " st_inf ", db, adOpenStatic, adLockOptimistic

الرابع : وهو نوع التزامن المطلوب للسجلات

هناك أربع أنواع للتزامن عند التعامل مع السجلات

حيث التزامن يقصد به نفس المفهوم للتزامن في الشبكات حيث إذا كان قام عميل أثناء عدم الاتصال بعرض بيانات وتم تعديلها أثناء عدم الاتصال فإن البيانات سيتم تحديثها بعد الاتصال مباشرة

AdLockReadOnly

حيث يتم فتح السجلات للقراءة فقط حيث لا يستطيع المستخدم تعديل السجلات أو تحديثها ويمكن لأكثر من مستخدم قراءة نفس السجلات في نفس الوقت .

AdLockPessimistic

وهو شكل صارم حيث يتم إغلاق السجل عندما يبدأ البرنامج بالتعامل معه حيث يمكن للمستخدمين الآخرين قراءة السجل ولا يمكنهم التعديل فيه أثناء تعديل البرنامج له وبعد تعامل البرنامج مع السجل يتم تحرير القفل ويستطيع المستخدمون الآخرون التعامل مع السجل وعرضه .

AdLockOptimistic

حيث يتم وضع أقفال لفترة صغيرة على السجلات أثناء تحديثها وهذا النوع الأكثر شيوعا والأفضل حيث يتم استخدامه بصورة دائمة ولكنه بطيء بعض الشيء .

AdLockBatchOptimistic

نفس النوع السابق ولكن يستخدم مع مجموعة السجلات المفصولة حيث يتم فصل السجلات للتعديل من العميل ثم يقوم بالاتصال بالملقم لحفظ التعديلات على قاعدة البيانات .

الحلقة الثامنة

العمليات الأساسية على السجلات

إضافة سجل جديد

لإضافة سجل جديد إلي قاعدة البيانات نكتب الكود التالي

```
If Txt1.Text ="" And Txt2.Text = "" And Txt3.Text = "" And
Txt4.Text = "" Then
      MsgBox("يجب إدخال القيم في مربعات النص")
Exit Sub
End if
rs.AddNew            جملة إضافة سجل جديد
    rs![st_no] = Val(Txt1)
    rs![st_name] = Txt2
    rs![address] = Txt3
    rs![b_date] = Txt4
    rs![phone] = Txt5
    rs.Update          تنفيذ الإضافة فعليا إلى الجدول في قاعدة البيانات وهي ضرورية
    cmdclear_Click
```

تعديل السجلات

لتعديل سجل معين نستخدم الكود التالي ونلاحظ هنا لأن الكائن ADO لا يستخدم الطريقة Edit كما هو موجود في الكائن DAO ولكن يقوم بالتعديل ثم حفظ التعديل بالطريقة Update

```
    rs![st_no] = Val(Txt1)
    rs![st_name] = Txt2
    rs![address] = Txt3
    rs![b_date] = Txt4
    rs![phone] = Txt5
   rs.Update
 End If
```

حذف سجل محدد باستخدام DELETE

لحذف سجل محدد يكتب الكود التالي

```
Dim x As Integer
 x = MsgBox "سيتم حذف البيانات"، vbYesNo("الحذف" ،
   If x = vbYes Then
      Do While Not rs.EOF
      If rs![st_no] = Val(text1) Then
       rs.Delete
    rs![st_no] = Val(Txt1)
    rs![st_name] = Txt2
    rs![address] = Txt3
    rs![b_date] = Txt4
    rs![phone] = Txt5
       Exit Sub
       End If
       rs.MoveNext
       Loop
   Else
    Txt1.SetFocus
    Exit Sub
  End If
```

حذف سجل محدد باستخدام DELETE مع جملة SQL

لحذف سجل باستخدام جملة SQL نكتب الكود التالي

```
Dim b As Integer
b = MsgBox  ("هل تريد بالتأكيد حذف السجل الحالي"  و vbMsgBoxRight +
vbYesNo و "حذف" )
If b = vbYes Then
db.Execute "delete from st_inf_info where st_no = " & Val(txt1)
rs.MoveFirst
rs.Fields.Refresh
     Txt1 = rs![st_no]
     Txt2 = rs![st_name]
     Txt3 = rs![address]
```

```
   Txt4 = rs![b_date]
   Txt5 = rs![phone]
Else
Txt1.SetFocus
End If
```

البحث عن سجل معين باستخدام جملة FIND

للبحث عن سجل باستخدام جملة FIND يتم كتابة الكود التالي :

يتم تعريف متغير نصي في أعلى النموذج منطقة تعريف المتغيرات كالتالي

Dim name as string

ثم يتم كتابة الكود التالي

```
Dim nam, s, y As String
y = InputBox("ادخل الاسم المراد البحث عنه", "بحث”)
 If Len(y) = 0 Then
   Exit Sub
 Else
   nam = y
   s = "st_name=' " + y" ' " +
   rs.Find s, 1, adSearchForward
     If rs.EOF Then
     MsgBox "الاسم غير موجود”
     Else
   Txt1 = rs![st_no[
   Txt2 = rs![st_name[
   Txt3 = rs![address[
   Txt4 = rs![b_date[
   Txt5 = rs![phone[
     End If
 End If
```

البحث باستخدام جملة DO WHILE

للبحث عن سجل باستخدام جملة DO WHILE نكتب الكود التالي :

```
Dim a As String
a = InputBox("ادخل اسم الطالب"، "بحث")
rs.MoveFirst
Do While Not rs.EOF'كرر لنهاية الجدول
If rs![emp_name] = a Then
Text1 = rs![St_no[
Text2 = rs![st_name[
Text3 = rs![ address[
Text4 = rs![b_date[
Text5 = rs![phone[
Exit Sub
End If
rs.MoveNext
Loop
```

وبقية العمليات مثل التحرك للسجل الأول والأخير والسابق والتالي نفس كود الكائن DAO بدون إختلاف

الدرس القادم سيكون برنامج عملي يشمل جميع العمليات باستخدام الكائن ADO وكذلك استخدام الأدوات DataList وأداة DataCombo وأداة DataGride وأداة MsflixGride

فأنتظرونا في الحلقة القادمة

الحلقة التاسعة

استخدام جمل SQL مع الفيجوال بيسك الجزء الأول

جمل sql ثمثل للمبرمج العمود الفقري الذي لا يمكن أن يستقيم برنامجه من دون استخدام هذه الجمل حيث يمكن باستخدام هذه الجمل التحكم الكامل بقاعدة البيانات وسنتحدث هنا عن العمليات الأساسية والتي تخدم المشروع من دون الخوض لبقية جمل SQL وهي كثيرة جدا فقط هنا عن عمليتين يتم استخدامها داخل الفيجوال بيسك بصورة كبيرة

١ – العمليات على الجداول (الإضافة .... الحذف ..... التعديل ...... الخ) حيث تستخدم هذه الجمل من خلال الكود التالي :

Db. Execute" SQL Statements"

٢ – عمليات عرض السجلات حيث يتم استخدامها من خلال الكود التالي

Rs.Open " SQL Statements"

فهيا بنا إلى العالم الممتع عالم الذي لا تجد فيه مستحيل عالم جمل SQL

أولا : العمليات الأساسية على الجداول

١ – جملة إنشاء جدول جديد Create Table

وهي جملة تستخدم لإنشاء جدول جديد داخل قاعدة البيانات وهي بالطبع أفضل الجمل في إنشاء جدول جديد وتحديد جميع خصائص هذا الجدول

الشكل العام للجملة يكتب على النحو التالي :

CREATE TABLE table_name (field1 type [(size)] [NOT NULL], field2 type [(size)] [NOT NULL] , CONSTRAINT multifieldindex)

شرح بارامترات الجملة :

كل ما وضع بين الحاصرتين [] هو اختياري أي يمكنك تحديده أو تركه ونبدأ ببارامترات الجملة

1 – create table جملة إنشاء جدول جديد

2 – table name اسم الجدول الجديد

3 – Field1 تحديد اسم الحقل

4 – Type وهو نوع بيانات الحقل وهو يشمل عدة أنواع أشهرها

Text نصي

Integer عدد صحيح صغير

Long عدد صحيح طويل

Memo مذكرة

Double عدد مزدوج ذات فاصلة عشرية

Datetime تاريخ ووقت

5 – size حجم الحقل ويكون للحقول النصية

6 – Not Null غير خالي أي يجب إدخال بيانات داخل هذا الحقل وإلا سيعطي رسالة خطأ

7 – CONSTRAINT تحديد المفتاح الأساسي للجدول حيث بعد كتابة الحقل ومواصفاته يتم تحديد هل الحقل مفتاح اساسي أم لا وهي اختيارية طبعا

مثال تنفيذي على الجملة

طبعا يجب ان يكون عندك قاعدة بيانات وأقترح استخدام آخر مثال تم وضعه في هذه السلسلة لتنفذ على التمارين التالية وهنا أنشأ زر أمر جديد واضغط عليه مرتين ثم أكتب الكود التالي

```
db.Execute "CREATE TABLE customers " _
    & "(FirstName TEXT, LastName TEXT, " _
    & "CustNO INTEGER CONSTRAINT MyFieldConstraint " _
    & "PRIMARY KEY);"
```

٢ – جملة تعديل الجدول Alter Table

وهذه الجملة تستخدم لتعديل بنية الجدول مثل إضافة حقول جديدة إلى الجدول وكذلك حذف حقول موجودة .

الشكل العام للجملة يكتب على النحو التالي :

ALTER TABLE table_name {ADD {COLUMN field type[(size)] [NOT NULL] DROP {COLUMN field I CONSTRAINT indexname} }

شرح بارامترات الجملة :

كل ما وضع بين الحاصرتين [] هو اختياري أي يمكنك تحديده أو تركه ونبدأ ببارامترات الجملة

1 – Alter table جملة إنشاء جدول جديد

2 – table_name اسم الجدول الجديد

3 – add column جملة إضافة حقل جديد إلى الجدول

4 – Field تحديد اسم الحقل

5 – Type وهو نوع بيانات الحقل وهو يشمل عدة أنواع أشهرها

Text نصي

Integer عدد صحيح صغير

Long عدد صحيح طويل

Memo مذكرة

Double عدد مزدوج ذات فاصلة عشرية

Datetime تاريخ ووقت

5 – size حجم الحقل ويكون للحقول النصية

6 – Not Null غير خالي أي يجب إدخال بيانات داخل هذا الحقل وإلا سيعطي رسالة خطأ

7 – CONSTRAINT تحديد المفتاح الأساسي للجدول حيث بعد كتابة الحقل ومواصفاته يتم تحديد هل الحقل مفتاح أساسي أم لا وهي اختيارية طبعا

8 – drop column لحذف حقل محدد من الجدول

مثال تنفيذي على الجملة

طبعا يجب ان يكون عندك قاعدة بيانات وأقترح استخدام آخر مثال تم وضعه في هذه السلسلة لتنفذ على التمارين التالية وهنا أنشأ زر أمر جديد واضغط عليه مرتين ثم أكتب الكود التالي

لإضافة حقل جديد

```
db.Execute "ALTER TABLE customers " _
  & "ADD COLUMN Salary CURRENCY;"
```

لحذف حقل موجود استخدم الجملة التالية

```
db.Execute "ALTER TABLE Customers " _
    & "DROP COLUMN Salary;"
```

٣ – جملة إضافة بيانات إلى الجدول Insert Into

حيث تستخدم هذه الجملة لإضافة بيانات إلى الجدول أي إضافة صف جديد إلى الجدول

الشكل العام للجملة يكتب على النحو التالي :

الشكل الأول : وحيث إدخال البيانات بصورة مباشرة إلى الجدول

INSERT INTO target [(field1[, field2[, ...]])]
VALUES (value1[, value2[, ...])

الشكل الثاني : إدخال البيانات إلى الجدول من خلال بيانات جدول آخر

INSERT INTO target [(field1[, field2[, ...]])] [IN externaldatabase]
SELECT field1[, field2[, ...]
FROM tableexpression

شرح بارامترات الجملة :

كل ما وضع بين الحاصرتين [] هو اختياري أي يمكنك تحديده أو تركه ونبدأ ببارامترات الجملة

1 – Insert Into جملة إدخال بيانات إلى الجدول

2 – target اسم الجدول المراد إدخال بيانات إليه

3 – Filed1,filed2,… أسماء الحقول المطلوب إدخال بيانات إليها

4 – VALUES جملة تحديد القيم المطلوب إدخالها

5 – value1, value2 القيم التي سيتم إدخالها إلى الجدول وهنا يجب أن تكون نفس نوع الحقول المحددة في البارامتر السابق

6 – IN وهي الجملة التي بعدها نكتب جملة بيانات الجدول الآخر

مثال تنفيذي على الجملة

أنشأ زر أمر جديد واضغط عليه مرتين ثم أكتب الكود التالي

مثال على الشكل الأول :

```
db.Execute " INSERT INTO Customers " _
  & "(FirstName,LastName, CustNo) VALUES " _
  & "('Harry', 'Washington', 10);"
```

مثال على الشكل الثاني:

```
db.Execute " INSERT INTO Northwind " _
    & "SELECT * " _
```

```
& "FROM Customers;"
```

٤ - عبارة التعديل في سجلات الجدول Update

وهي العبارة التي تستخدم بتعديل البيانات داخل الجدول حيث يمكن تعديل كمية كبيرة من البيانات مرة واحدة

الشكل العام للجملة يكتب على النحو التالي :

```
UPDATE table
SET col1 = Value, col2 = value2 , …..
WHERE criteria ;
```

شرح بارامترات الجملة :

1 – Update جملة التعديل

2 – Table اسم الجدول المراد تعديل بياناته

3 – SET جملة تحديد القيمة الجديدة

4 – Where جملة تحرير الشرط

5 – criteria الشرط المطلوب تنفيذه

مثال تنفيذي على الجملة

```
db.Execute "UPDATE Customers " _
    & "SET CustNO = 5 " _
    & "WHERE FirstName = Ahmed ;"
```

مثال ليس للتنفيذ

```
db.Execute "UPDATE Customers " _
    & "SET Country = ' Germany' " _
    & "WHERE Country = 'w' Or country = 'E' ;"
```

٥ – جملة حذف بيانات من الجدول Delete

لحذف بيانات من الجدول حيث تقوم بحذف صف أو أكثر من الجدول ويمكن استخدامها تحت شرط معين

الشكل العام للجملة يكتب على النحو التالي :

```
DELETE
FROM table_name
WHERE criteria
```

<u>شرح بارامترات الجملة :</u>

1 – DELETE   جملة الحذف

2 – Table_name اسم الجدول المراد حذف بياناته

3 – Where   جملة تحرير الشرط

4– criteria الشرط المطلوب تنفيذه

<u>مثال تنفيذي على الجملة</u>

```
db.Execute "DELETE * FROM " _
    & "Customers WHERE firstname = 'ahmed';"
```

<u>ملاحظة هامة</u>

عند استخدام تعبير يشير إلى مربع نص معين مثل

إذا كان النص يحتوي على بيانات نصية تكتب الجملة كالتالي :

"Where firstName = ' " + text1.text + " ' "

إذا كان مربع النص يحتوي على قيمة رقمية تكتب الجملة كالتالي :

"where CustNo = " & text1.text

<u>٦ – جملة حذف جدول Drop table</u>

وتستخدم هذه الجملة لحذف الجدول بجميع محتوياته من قاعدة البيانات ولا يمكنك استعادة الجدول بعد حذفه من قاعدة البيانات .

<u>الشكل العام للجملة يكتب على النحو التالي :</u>

```
DROP TABLE table_name
```

<u>شرح بارامترات الجملة :</u>

1 – Drop Table   جملة حذف الجدول

2 – Table_name اسم الجدول المراد حذف بياناته

<u>مثال تنفيذي على الجملة</u>

```
db.Execute "DROP TABLE Customers "
```

الحلقة العاشرة

الجزء الثاني من استخدام جمل SQL مع الفيجوال بيسك

استخدام جملة العرض Select

حيث نستطيع استخدام جمله select مع الفيجوال عند فتح جدول حيث يتم استخدامها بطريقتين

الأولى : حيث يتم تعريف متغير نصي ثم نضع فيه جملة Select ثم نقوم بعد ذلك بفتح الجملة كالتالي :

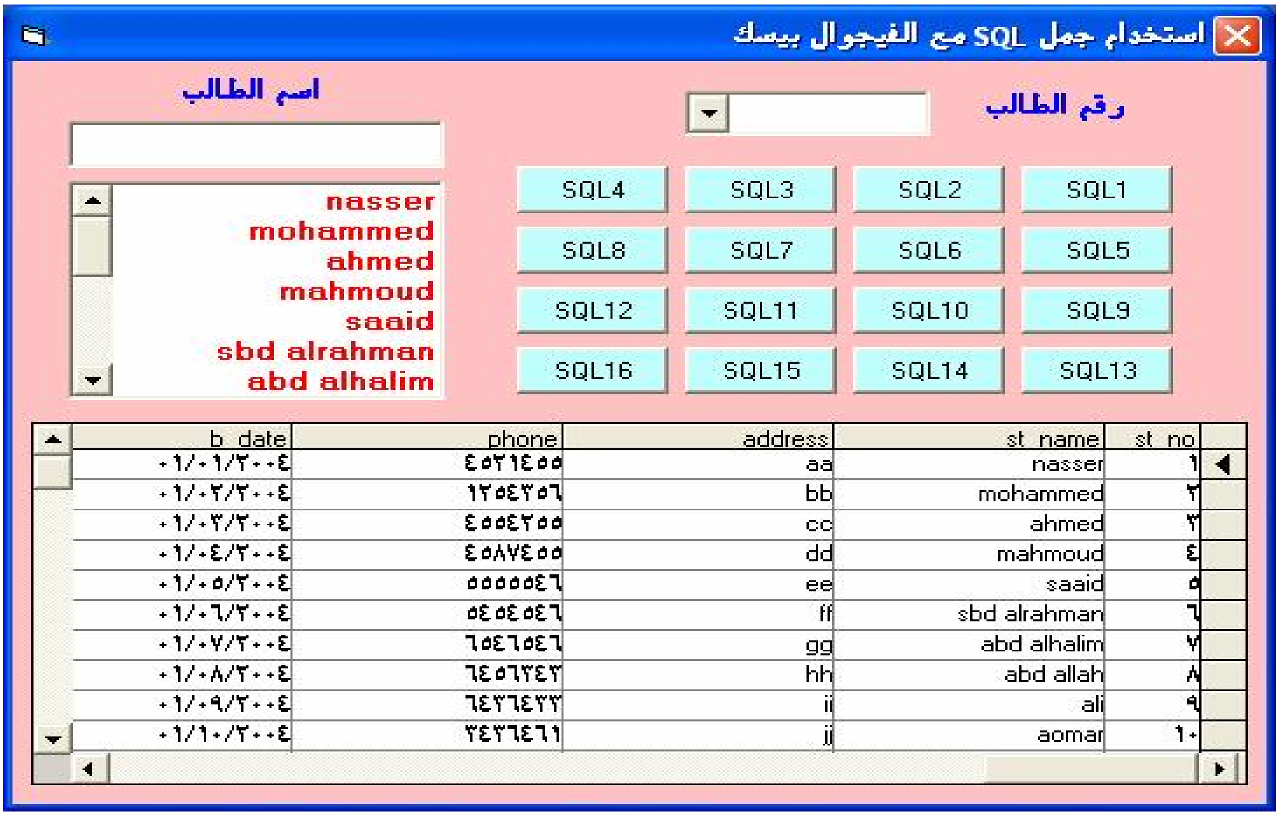
```
Dim sql As String
sql = "Select ........"
If rs.State = adStateOpen Then rs.Close
rs.Open sql, db, adOpenDynamic, adLockOptimistic
```

ثم بعد ذلك يتم استخدام المتغير Rs الذي يحمل ناتج تنفيذ جملة SQL في أي مكان تريده

الثانية : حيث يتم استخدام الجملة مباشرة عند فتح الجدول كالتالي

```
rs1.Open "select …. " , db, adOpenDynamic, adLockOptimistic
```

وهنا سنستخدم نفس الأسلوب الذي استخدمناه في الجزء الأول حيث سنعرض الجملة ونقوم بشرحها ثم نعطي مثال عملي عليها قم بتنزيل المثال المرفق ويا ريت تقوم بحفظة باسم جديد ثم تقوم بحذف الكود الموجود حتى تكتب الكود أنت بنفسك



الشكل العام لجملة Select

SELECT col1, col2, col3,……
From tab1,tab2,…
WHERE conditions
[GROUP BY...]
[HAVING...]
[ORDER BY...]

هذه هي الصورة العامة لجملة Select والتي من الممكن أن تأخذ معنا عـدد مـن الحلقـات فتحملونا في هذه الحلقات التي أظن أنها هامة لي مبرمج يريد أن يكون مبدعا وسـنبدأ مـع الجملة البسط لجملة Select ثم نمر معها حتى نصل إلى نهاية هذا الشكل ولنبدأ بشرح مبسط لبارامترات الجملة

١ – select وهي جملة العرض لمجموعة من العمدة من جدول أو أكثر تحت شرط أو مجموعة من الشروط

٢ – col1,col2,col3 هي أسماء الحقول المراد استخراجها

٣ – From وهي جملة تحديد الجداول

٤ – tab1,tab2 هي أسماء الجداول التي يتم عرض الحقول منها

٥ – where جملة تجهيز الشرط

٦ – Conditions الشروط المطلوب تنفيذها على الحقول

٧ – Group by هي جملة التجميع

٨ – having هي جملة تقليص التجميع المستخدم بجملة Group By

٩ – Order By هي جملة الترتيب حسب حقل أو مجموعة من الحقول

هذا شرح سريع لبارامترات الجملة وهيا بنا شمر حتى نخوض في التفاصيل

١ – استعادة أعمدة محددة من الجدول وهي أبسط عبارات جملة Select والتي تكتب كالتالي

Select Fields From Tables

حيث Fields تمثل الحقول المراد إظهارها Tables تمثل الجداول التي تحوي هذه الحقول

مثال تطبيقي ١

لعرض مجموعة حقول محددة من جدول

اضغط على الزر SQL1 مرتين ثم اكتب الكود التالي

```
Dim sql As String
sql = "Select St_no,st_name,Address From St_inf"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

٢ – لعرض جميع الحقول من جدول محدد حيث يتم كتابة الكود التالي :

```
Select  * from  table
```

حيث تعبر * عن جميع الحقول بالجدول

مثال تطبيقي ٢

لعرض مجموعة حقول محددة من جدول

اضغط على الزر SQL2 مرتين ثم اكتب الكود التالي

```
Dim sql As String
sql = "Select  *  From St_inf "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

٣ – لعرض حقول من جداول مختلفة نكتب الجملة التالية

```
Select  tab1.col1,tab2.col2……
From tab1,tab2,…..
```

مثال تطبيقي ٣

لعرض مجموعة حقول محددة من جدول

اضغط على الزر SQL3 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select
st_inf.st_no,st_inf.st_name,st_inf.address,st_level.level,st_level.class
From St_inf,st_level"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

وهنا إذا لم تضع شرطا معينا فسييتم تكرار السجلات بمعنى أن كل سجل من الجدول الأول سيظهر له كل سجلات الجدول الثاني ولذلك إذا أردت عرض عدد معين من السجلات يمكنك استخدام الجملة التالية

٤ – عرض عدد محدد من السجلات باستخدام جملة Top n حيث تمثل ى عدد السجلات المراد عرضها والجملة تكتب كالتالي :

```
Select  Top n tab1.col1,tab2.col2……
From tab1,tab2,…..
```

حيث يتم عرض العدد المحدد من السجلات

مثال تطبيقي ٤

لعرض مجموعة حقول محددة من جدول

اضغط على الزر SQL4 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select top 17
st_inf.st_no,st_inf.st_name,st_inf.address,st_level.level,st_level.class
From St_inf,st_level"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

- **وهنا يمكنك استخدام جملة All بدلا من Top n لعرض جميع السجلات مرة ثانية استبدل فقط كلمة Top n بكلمة all سيتم عرض جميع السجلات**
- **كذلك يمكنك استخدام كلمة DISTINCT لمنع ظهور تكرار السجلات مثل الجملة التالية :**

```
SELECT DISTINCT
LastName
FROM Employees;
```

لمنع تكرار الأسماء المتشابهة في الأسم الأخير

- **ولمنع التكرار من خلال عدة جداول مرتبطة معا يمكنك استخدام البارامتر DISTINCTROW مثل عند عرض العملاء وطلبياتهم يمكن أن يكون للعميل أكثر من طلبيه وأنت تريد عرض طلبيه واحدة لكل عميل فنستخدم المثال التالي**

```
SELECT DISTINCTROW CompanyName
FROM Customers INNER JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
ORDER BY CompanyName;
```

- كذلك يمكنك تغيير اسماء الحقول المعروضة باستخدام جملة AS حيث يتم كتابة الجملة كالتالي :

```
Select col1 as new1,col2 as new2
From tab
```

مثال تطبيقي ٥

لعرض حقول محددة وتغيير هذه الحقول بأسماء جديدة

اضغط على الزر SQL5 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select st_no as [رقم الطالب],st_name as [اسم الطالب],address as
[العنوان]  From St_inf "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

استخدام جملة Where

والتي تستخدم لإدراج الشروط وهنا يمكنك استخدام عدة إشكال من الشروط سوف نتناول جميع الأشكال من الشروط

أولا : شروط المقارنة وتشمل العلامات المعروفة في المقارنة مثل

= أو > أو < أو >= أو <= أو <> (لا يساوي) حيث يتم كتابة الجملة كالتالي :

```
Select * from tab1
Where col { } value
```

حيث يتم وضع معامل المقارنة مكان القوسين { } ويكون Value هي قيمة المقارنة التي ممكن أن تكون رقم أو قيمة مدخلة من مربع نص أو قيمة مدخلة من أي أداة أخري

مثال تطبيقي ٦

لعرض جميع حقول الجدول بحيث أن رقم الطالب أكبر من ٦ مثلا

اضغط على الزر SQL6 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select  * From St_inf where st_no > 6"
If rs1.State = adStateOpen Then rs1.Close
```

```
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

صور أخري يمكنك تجريبها

Select * from st_inf where st_no <=6

لعرض مجموعة سجلات بحيث رقم الطالب أقل من أو يساوي ٦

Select * from st_inf where b_date >#1/6/2004#

لعرض مجموعة سجلات بحيث تاريخ الميلاد أكبر من أو يساوي 1/6/2004

ثانيا : استخدام المعاملات المنطقية وهي ثلاث

- **And وهي تستخدم لربط شرطين بحيث يكون الناتج صحيح إذا تحقق الشرطين معا**
- **Or وهي اربط شرطين بحيث يكون ناتج صحيح إذا تحقق أحد الشرطين**
- **Not وهو شرط لنفي الجملة عكس النتائج وتستخدم مع جمل أخرى كما سيرد لاحقا**

وتكتب الجملة كالتالي :

```
Select * from tab
Where col >= n1 [and /or ] col <n2
```

مثال تطبيقي ٧

لعرض جميع حقول الجدول بحيث أن رقم الطالب أكبر من ٦ مثلا واقل من ١٥

اضغط على الزر SQL7 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select  * From St_inf where st_no > 6 and st_no <15"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

ثالثا : استخدام الجملة Between .... And أو not between ...and

وتستخدم لعرض البيانات المحصورة بين قيمتين ممكن أرقام وممكن تواريخ وتكتب الجملة كالتالي :

```
Select * from tab
Where col Between n1 and n2
```

مثال تطبيقي ٨

لعرض جميع حقول الجدول بحيث أن رقم الطالب محصور بين ٦ و ١٥ مع الأخذ في الاعتبار أن الرقمين ٦ و ١٥ قيمة البداية والنهاية ستظهر في العرض

اضغط على الزر SQL8 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select  * From St_inf where st_no between 6 and 15"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

رابعا : استخدام العامل ضمن IN

حيث تستخدم لعرض القيم الموجودة داخل قوس وتكتب كالتالي :

```
Select * from tab
Where col in ( n1,n2,n3,….)
```

مثال تطبيقي ٩

لعرض جميع حقول الجدول بحيث أن رقم الطالب له القيم داخل القوس

اضغط على الزر SQL9 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select  * From St_inf where st_no In(4,6,8,10,12,14)"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

نكتفي بهذا في هذه الحلقة ونكمل إن شاء الله في الحلقة القادمة

هانت يا شباب أوشكنا على النهاية إن شاء الله ونتم الموضوع بإذن الله تعالى

الحلقة الحادية عشر

الجزء الثالث من استخدام جمل Sql مع الفيجوال بيسك

خامسا : استخدام العامل Like

حيث تستخدم لعرض القيم تبدا بحرف او تنتهي بحرف أو تحتوي على حرف مثل عندما تريد عرض أسماء الطلاب التي تحتوي على كلمة محمد مثلا :

الشكل العام للجملة يكتب كالتالي :

Select * from tab
Where col like string

حيث تمثل **string** قيمة النص الذي نريد عرضه وهو كالتالي :

أولا : إذا كنت تستخدم قاعدة البيانات أكسس فيكون النص كالتالي

الرمز	الشرح	مثال
*	تحل محل مجموعة من الحروف	- **A*** لعرض كل الأسماء التي تبدأ بحرف **A** - **A*** لعرض السماء التي تنتهي بحرف **A** - ***A*** لعرض الأسماء التي تحتوي على حرف ويمكن بدل الحرف طبعا وضع كلمة
?	تحل محل حرف واحد فقط	**S?d** : فيتم عرض **sad** أو **sed** وهكذا
[]	حيث يعرض حرف من الحروف داخل القوسين مثل جملة **IN**	**Ahmed al[I,e,y]** حيث يعرض **Ahmed ali** أو **Ahmed ale** أو **Ahmed aly**
[!]	حيث يعرض حرف من غير الحروف داخل القوسين مثل	***c[!q]** تعني عرض الأسماء التي تنتهي بالحرفين **c** مع أي حرف غير **q**
[-]	عرض حروف تقع ضمن المدى المحدد بين القوسين	***[a-d]** عرض كل السماء التي تبدا بالحروف من **a** الى **d** بالترتيب طبعا ولا تفرق الحروف الكبيرة والصغيرة في البحث
#	تقابل رقم واحد	**C1##** لعرض القيم **c100** و**c120** وهكذا

ثانيا : إذا كنت تستخدم قاعدة البيانات Sql server فيكون النص كالتالي :

الرمز	الشرح	مثال
%	تحل محل مجموعة من الحروف	- A% لعرض كل الأسماء التي تبدأ بحرف A - %A لعرض السماء التي تنتهي بحرف A - %A% لعرض الأسماء التي تحتوي على حرف ويمكن بدل الحرف طبعا وضع كلمة
-	الشرطة السفلية تحل محل حرف واحد فقط	d_S : فيتم عرض sad أو sed وهكذا
[]	حيث يعرض حرف من الحروف داخل القوسين مثل جملة IN	Ahmed al[I,e,y] حيث يعرض Ahmed ali أو Ahmed ale أو Ahmed aly
[^]	حيث يعرض حرف من غير الحروف داخل القوسين مثل	%c[!q] تعني عرض الأسماء التي تنتهي بالحرفين c مع أي حرف غير q
[-]	عرض حروف تقع ضمن المدى المحدد بين القوسين	[a-d]% عرض كل السماء التي تبدأ بالحروف من a إلى d بالترتيب طبعا ولا تفرق الحروف الكبيرة والصغيرة في البحث
#	تقابل رقم واحد	C1## لعرض القيم c100 وc120 وهكذا

وهنا ملاحظة ان الرموز مع الكائن ado هي المستخدمة مع sql server حتى ولو كنت تستخدم access

مثال تطبيقي ١٠

لعرض جميع حقول الجدول بحيث أن اسم الطالب يحتوي على حرف m

اضغط على الزر SQL10 مرتين ثم اكتب الكود التالي :

```
Dim sql As String
sql = "Select  * From St_inf where st_name like *m*"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

مثال تطبيقي ١١

لعرض جميع الأسماء في قائمة اسم الطالب عندما نكتب أي حرف في مربع النص الذي فوقها اضغط على مربع النص الذي فوق list box مرتين ثم اختار الحث Change لمربع النص ثم اكتب الكود التالي :

```
Dim sql
sql = "select * from st_inf where st_name like '%" + txt6.Text + "%'"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set dblist1.RowSource = rs1
dblist1.ListField = "st_name"
dblist1.Refresh
```

سادسا : جملة الترتيب Order By

حيث تستخدم لترتيب البيانات المعروضة حسب حقل أو أكثر :

الشكل العام للجملة يكتب كالتالي :

```
Select * from tab
Where condition
Order by col1,col2,…[Asc,Desc]
```

حيث يمكنك ترتيب البيانات حسب حقل أو أكثر ترتيبا تصاعديا وهو الوضع الافتراضي للترتيب أو ترتيبا تنازليا حيث يلزمك استخدام البارامتر Desc

مثال تطبيقي ١٢

لعرض جميع الأسماء مرتبة ترتيبا تصاعديا حسب اسم الطالب

اضغط على الزر Sql11 مرتين ثم اكتب الكود التالي :

```
Dim sql
sql = "select * from st_inf where st_no>6 order by st_name"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

الحقول المحسوبة داخل جملة Select

حيث يمكنك إجراء عمليات حسابية على البيانات المعروضة وهذه العمليات تكون مفيدة جدا في حالات لا تجد إلا الحقول المحسوبة مثل إيجاد مجموع البيانات داخل حقل أو إيجاد أكبر قيمة .....

الدوال التي يمكن استخدامها في جملة select هي :

١ – دالة المجموع Sum()

حيث يمكنك استخدامها لإيجاد مجموع القيم الموجودة في حقل معين

مثال تطبيقي ١٣

مثلا لحساب مجموع القيم الموجودة في الحقل st_no وهو الحقل الرقمي الموجود بالجدول

اضغط على الزر Sql12 مرتين ثم اكتب الكود التالي :

```
Dim sql
sql = "select sum(st_no) from st_inf "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
text1.text = rs1.Fields(0)
```

٢ – دالة أكبر قيمة Max()

حيث يمكنك استخدامها لإيجاد أكبر القيم الموجودة في حقل معين

مثال تطبيقي ١٤

مثلا لحساب أكبر القيم الموجودة في الحقل st_no وهو الحقل الرقمي الموجود بالجدول

اضغط على الزر Sql13 مرتين ثم اكتب الكود التالي :

```
Dim sql
sql = "select max(st_no) from st_inf "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
text1.text = rs1.Fields(0)
```

٣ – دالة أقل قيمة Min()

حيث يمكنك استخدامها لإيجاد اقل القيم الموجودة في حقل معين

مثال تطبيقي ١٤

مثلا لحساب اقل القيم الموجودة في الحقل st_no وهو الحقل الرقمي الموجود بالجدول

اضغط على الزر Sql14 مرتين ثم اكتب الكود التالي :

```
Dim sql
sql = "select min(st_no) from st_inf "
If rs1.State = adStateOpen Then rs1.Close
```

```
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
text1.text = rs1.Fields(0)
```

٤ - دالة المتوسط Avg()

حيث يمكنك استخدامها لإيجاد متوسط القيم الموجودة في حقل معين

مثال تطبيقي ١٥

مثلا لحساب مجموع القيم الموجودة في الحقل st_no وهو الحقل الرقمي الموجود بالجدول

اضغط على الزر Sql15 مرتين ثم اكتب الكود التالي :

```
Dim sql
sql = "select avg(st_no) from st_inf "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
text1.text = rs1.Fields(0)
```

٥ – دالة عدد السجلات Count()

حيث يمكنك استخدامها لإيجاد عدد السجلات الموجودة في حقل معين حيث يتم تجاهل القيم الخالية إلا إذا استخدمت علامة *

مثال تطبيقي ١٦

مثلا لحساب عدد السجلات الموجودة في الحقل st_no وهو الحقل الرقمي الموجود بالجدول

اضغط على الزر Sql16 مرتين ثم اكتب الكود التالي :

```
Dim sql
sql = "select count(st_no) from st_inf "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
text1.text = rs1.Fields(0)
```

٦ – يمكنك إجراء عملية حسابية على حقل معين

مثل ضرب الحقل رقم الطالب في ١٠

مثال تطبيقي ١٧

مثلا لضرب الحقل st_no في العدد ١٠

اضغط على الزر Sql17 مرتين ثم اكتب الكود التالي :

```
Dim sql
sql = "select st_no,st_no*10 as account from st_inf "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
set datagrid1.datasource = rs1
```

الحلقة الثانية عشر

الجزء الرابع من استخدام جمل Sql مع الفيجوال بيسك

جملة التجميع Group By

تستخدم لتجميع النتائج الى مجموعات حسب حقل من الحقول حيث يمكنك مثلا ايجاد المجموعات الفرعية لعدة اقسام باستخدام جملة Group By

الصورة العامة للجملة تكتب كالتالي :

```
Select * from tab
Where condition
Group By col
Order by col1,col2,…[Asc,Desc]
```

نعطي مثال تطبيقي سريع

مثلا إذا كان عندنا رواتب الموظفين للشركة والتي بها مجموعة أقسام ونريد ايجاد مجموع الرواتب لكل قسم من الأقسام افترض عندنا ثلاث أقسام قسم الحسابات Account – قسم التدريب Training وقسم المبيعات Sales

اضغط على الزر SQL18 مرتين ثم اكتب الجملة التالية

```
Dim sql
sql = "select dep,sum(salary) as the_sum,avg(salary) as
the_averag,count(salary) as the_aacount from emp group by dep "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

جملة تقليص التجميع Having

تستخدم لتقليص المجموعات مثل إيجاد أكبر مجموع من المثال السابق

الصورة العامة للجملة تكتب كالتالي :

```
Select * from tab
Where condition
Group By col
Having col
Order by col1,col2,…[Asc,Desc]
```

نعطي مثال تطبيقي سريع

مثلا إذا كان عندنا رواتب الموظفين للشركة والتي بها مجموعة أقسام ونريد ايجاد المجموعات الأكبر من ١٢٠٠٠

مثال تطبيقي ١٩

اضغط على الزر SQL19 مرتين ثم اكتب الجملة التالية

```
Dim sql
sql = "select dep,sum(salary) as the_sum,avg(salary) as
the_averag,count(salary) as the_aacount from emp group by dep
having sum(salary)>12000 and avg(salary)>3000"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

ربط الجداول

وهي ميزة هامة جدا من مميزات قاعدة البيانات أنك يمكنك عرض بيانات من اكثر من جدول ولكن أولا يجب أن يكون بين الجداول علاقة ربط معينة حتى يمكنك استخراج المعلومات من هذه الجداول ولمن لا يعرف عن العلاقات هو يجب وجود حقل رقمي مكرر في الجدولين يطلق على الحقل في الجدول الأول المفتاح الأساسي ويطلق عليه في الجدول الثاني المفتاح الأجنبي حيث يتم الشرط من خلال تساوي قيم هذين الحقلين

جمل الوصل

١ – الوصل باستخدام جملة Where

٢ – الوصل باستخدام جملة Join

أولا : الوصل باستخدام جملة Where

حيث تكتب جملة الربط كالتالي

```
Select tab1.col1, tab2.col2
From tab1,tab2
Where tab1.col = tab2.col
```

ونعطي مثلا عمليا حيث نعتبر الجدولين st_info والجدول st_level

الجدول الأول به الحقول st_no,st_name,address,phone,b_date,

والجدول الثاني به الحقول التالية st_no,level,class مثلا

مثال تطبيقي ٢٠

إذا أردنا مثلا عرض رقم الطالب والمستوى والفصل من الجدولين بحيث أن رقم الطالب متساوي

اضغط على الزر SQL20 واكتب الكود التالي

```
Dim sql
sql = "select st_inf.st_no,st_level.level,st_level.class from st_inf,st_level
 where st_inf.st_no = st_level.st_no and st_inf.st_no>8 "
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

ثانيا : الوصل باستخدام الجملة Join

وهي تأتي على عدة أشكال

الشكل الأول : جملة Inner join

وهي الوضع الافتراضي إن لم تحدد نوع الصلة وتسمى صلة داخلية حيث تعيد جميع السجلات المتطابقة من الجدولين وتهمل السجلات الغير متطابقة

وهي تكتب كالتالي :

```
Select tab1.col1, tab2.col2
From tab1 Inner join tab2 on tab1.col = tab2.col
Where condition
Order by col,…
Group by col ,….
```

مثال تطبيقي ٢١

اضغط على الزر SQL21 مرتين فمثلا إذا أردنا مثلا عرض رقم الطالب والمستوى والفصل من الجدولين بحيث أن رقم الطالب متساوي

```
Dim sql
sql = "select st_inf.st_no,st_level.level,st_level.class from st_inf inner
 join st_level on st_inf.st_no = st_level.st_no"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

الشكل الثاني : جملة left join

تسمى صلة يسارية (ليست شيوعية) ولكن من اليسار حيث تعيد جميع السجلات للجدول اليساري (الثانوي) والمطابقة لها من الجدول اليميني (الأساسي)

وهي تكتب كالتالي :

```
Select tab1.col1, tab2.col2
From tab1 Left join tab2 on tab1.col = tab2.col
Where condition
Order by col,…
Group by col ,….
```

مثال تطبيقي ٢٢

اضغط على الزر SQL22 مرتين فمثلا إذا أردنا مثلا عرض رقم الطالب والمستوى والفصل من الجدولين بحيث أن رقم الطالب متساوي بحيث يعرض كل سجلات الثاني والمتساوي من الأول

```
Dim sql
sql = "select st_inf.st_no,st_level.level,st_level.class from st_inf Left join st_level on st_inf.st_no = st_level.st_no where st_inf.st_no>10"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

الشكل الثالث : جملة right join

تسمى صلة يمينية ولكن من اليمين حيث تعيد جميع السجلات للجدول اليميني (الأساسي) والمطابقة لها من الجدول اليساري (الثانوي)

وهي تكتب كالتالي :

```
Select tab1.col1, tab2.col2
From tab1 Right join tab2 on tab1.col = tab2.col
Where condition
Order by col,…
Group by col ,….
```

مثال تطبيقي ٢٣

اضغط على الزر SQL23 مرتين فمثلا إذا أردنا مثلا عرض رقم الطالب والمستوى والفصل من الجدولين بحيث أن رقم الطالب متساوي بحيث يعرض كل سجلات الأول مع المتساوي فقط من الثاني

```
Dim sql
sql = "select st_inf.st_no,st_level.level,st_level.class from st_inf Right
join st_level on st_inf.st_no = st_level.st_no where st_inf.st_no>10"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

الشكل الرابع : جملة Full join

تسمى صلة كاملة حيث تعيد جميع السجلات من الجدولين

وهي تكتب كالتالي :

```
Select tab1.col1, tab2.col2
From tab1 full join tab2 on tab1.col = tab2.col
Where condition
Order by col,…
Group by col ,….
```

مثال تطبيقي ٢٤

اضغط على الزر SQL24 مرتين فمثلا إذا أردنا مثلا عرض رقم الطالب والمستوى والفصل من الجدولين بحيث أن رقم الطالب متساوي

```
Dim sql
sql = "select st_inf.st_no,st_level.level,st_level.class from st_inf full
join st_level on st_inf.st_no = st_level.st_no where st_inf.st_no>10"
If rs1.State = adStateOpen Then rs1.Close
rs1.Open sql, db, adOpenDynamic, adLockOptimistic
Set DataGrid1.DataSource = rs1
```

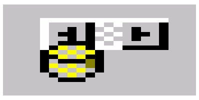
بهذا نكون قد انهيا أهم جمل SQL المستخدمة مع الفيجوال بيسك

والى لقاء مع الجزء الأخير من هذه السلسلة وهو Data report

الحلقة الثالثة عشر

استخدام : أداة قاعدة البيانات ADODBC(Ado Data Base Control)

وشكلها على صندوق الأدوات كالتالي



الخطوة الأولى :

هذه الأداة من الأدوات التي يجب إضافتها إلى صندوق الأدوات ويتم إضافتها كالتالي

افتح قائمة PROJECT واختار منها COMPONENTS من الشاشة التي تخرج أختار العنصر التالي

MICROSOFT ADO DATA CONTROL 6.0 (OLEDB)

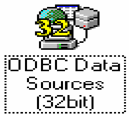
ضع أمامها علامة صح ثم بعد ذلك يتم وضع الأداة على النموذج

وتوضع على النموذج كما ترى عليها السهم الأبيض على النموذج في الشكل التالي

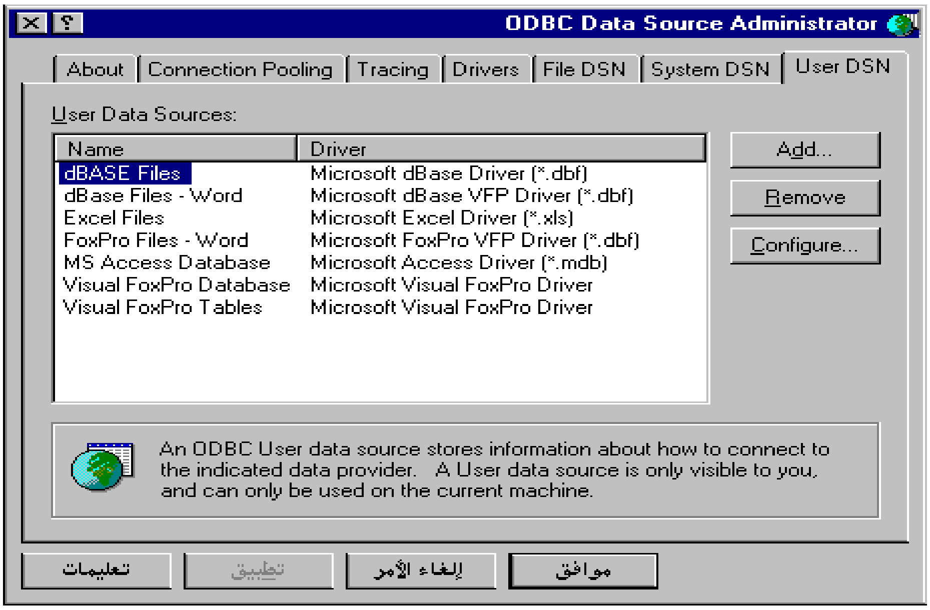


الخطوة الثانية :

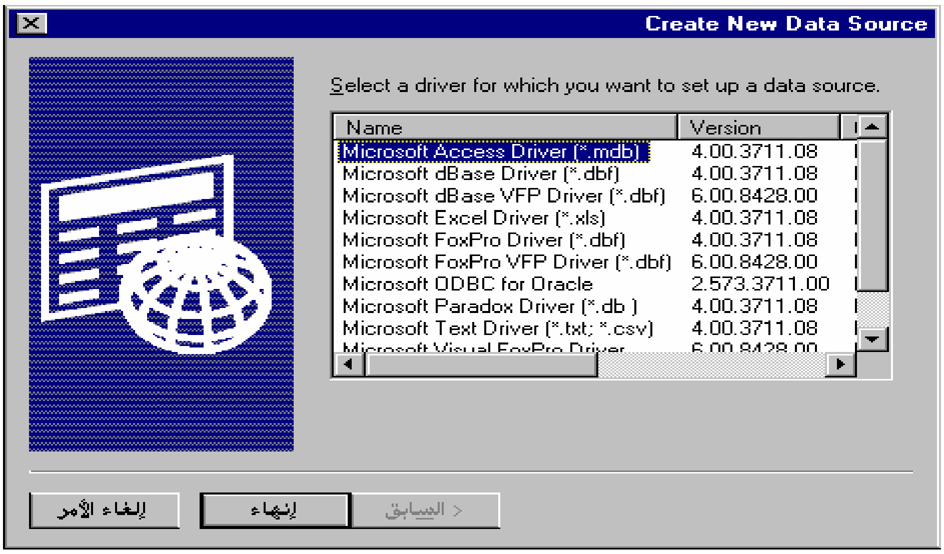
ضبط خصائص الأداة وإنشاء اتصال مع قاعدة البيانات المراد استخدامها ويتم ذلك كالتالي

١ - افتح لوحة التحكم واضغط مرتين على الأية 

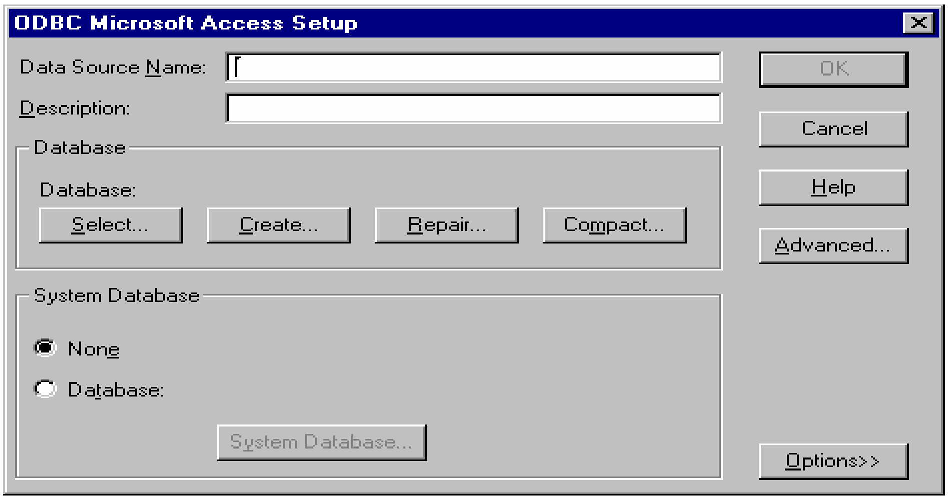
٢ – تظهر الشاشة التالية



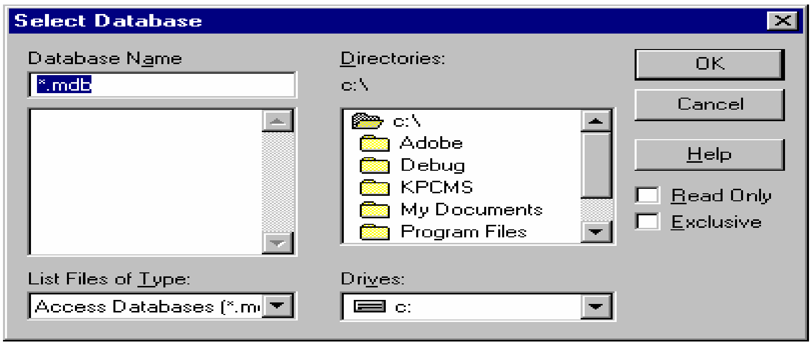
وهذه الشاشة تستخدم لتحديد برنامج قواعد البيانات الذي سيتم التعامل معه اضغط زر ADD تظهر الشاشة التالية لاختيار البرنامج



نحدد البرنامج الذي تم إنشاء قاعدة البيانات بواسطته وليكن **microsoft access driver** ثم اضغط على زر إنهاء فتظهر الشاشة التالية

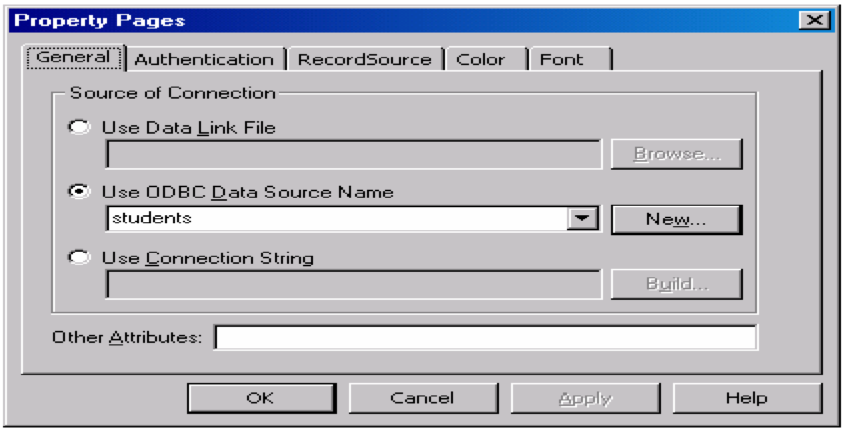


من هذه الشاشة يتم وضع اسم للإتصال بقاعدة البيانات في خانة **Data source name** اكتب اسم الإتصال وليكن **student** ثم اضغط على زر **select** لتحديد قاعدة البيانات المراد الإتصال بها من الشاشة التالية



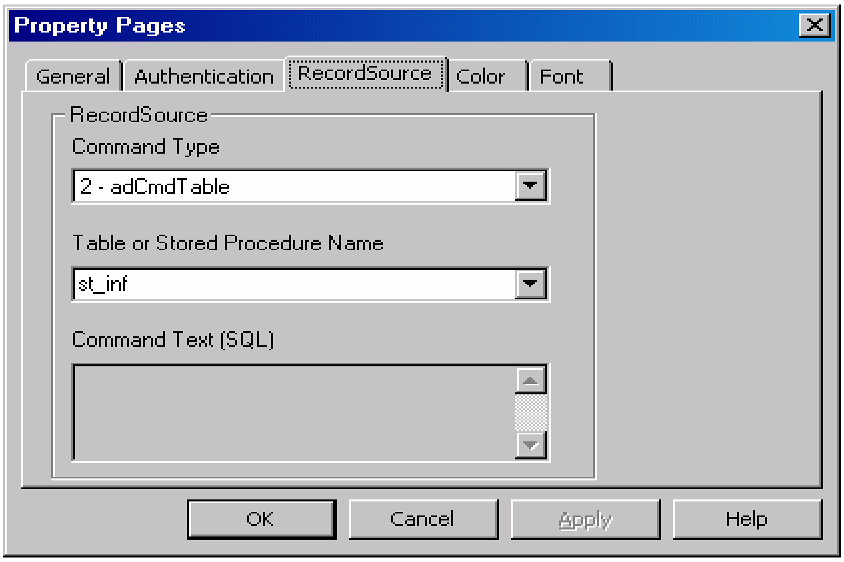
من هذه الشاشة حدد مسار قاعدة البيانات المراد الإتصال بها ثم اضغط زر **OK**

بعد ذلك أذهب إلى برنامج الفيجوال واضغط على أداة قواعد البيانات **ADODBC** واختار من القائمة **ADODC properties** تظهر الشاشة التالية



من باب **general** نحدد الخيار **Use odbc data source** ونختار من مربع السرد الاسم الذي تم تحديده للاتصال

ومن باب **Recourd source**



تحت **command type** اختار **Adcmd table** لتحديد الجدول المراد ربط الأداة به فتظهر في الأسفل مربع سرد لاختيار اسم الجدول المراد استخدامه ثم اضغط على زر **ok**
وهكذا تم ربط الأداة

وبعدها يتم التعامل مع قاعدة البيانات من خلال التعليمات
حيث يتم بنفس التعليمات السابقة ولكن التغيير الذي سيتم فقط هو تغيير كل جملة تحتوي على **RS** إلى الجملة التالية **DATA.RECORDSET** كالتالي على سبيل المثال

RS![ST_NO] = VAL(TEXT1)

يتم تغييرها إلى

ADO1.RECORDSET![ST_NO] = VAL(TEXT1)

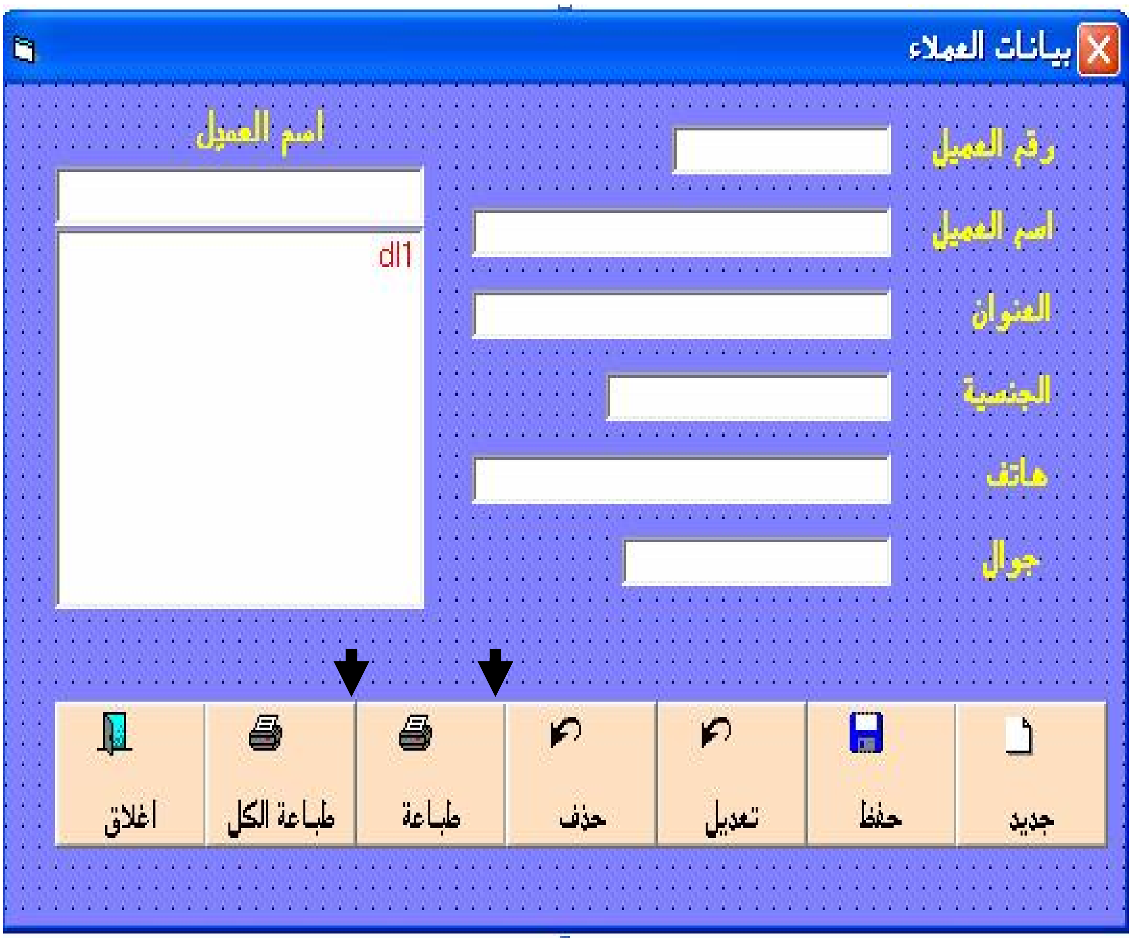
وهكذا بقية التعليمات في العمليات على السجلات

الحلقة الرابعة عشر

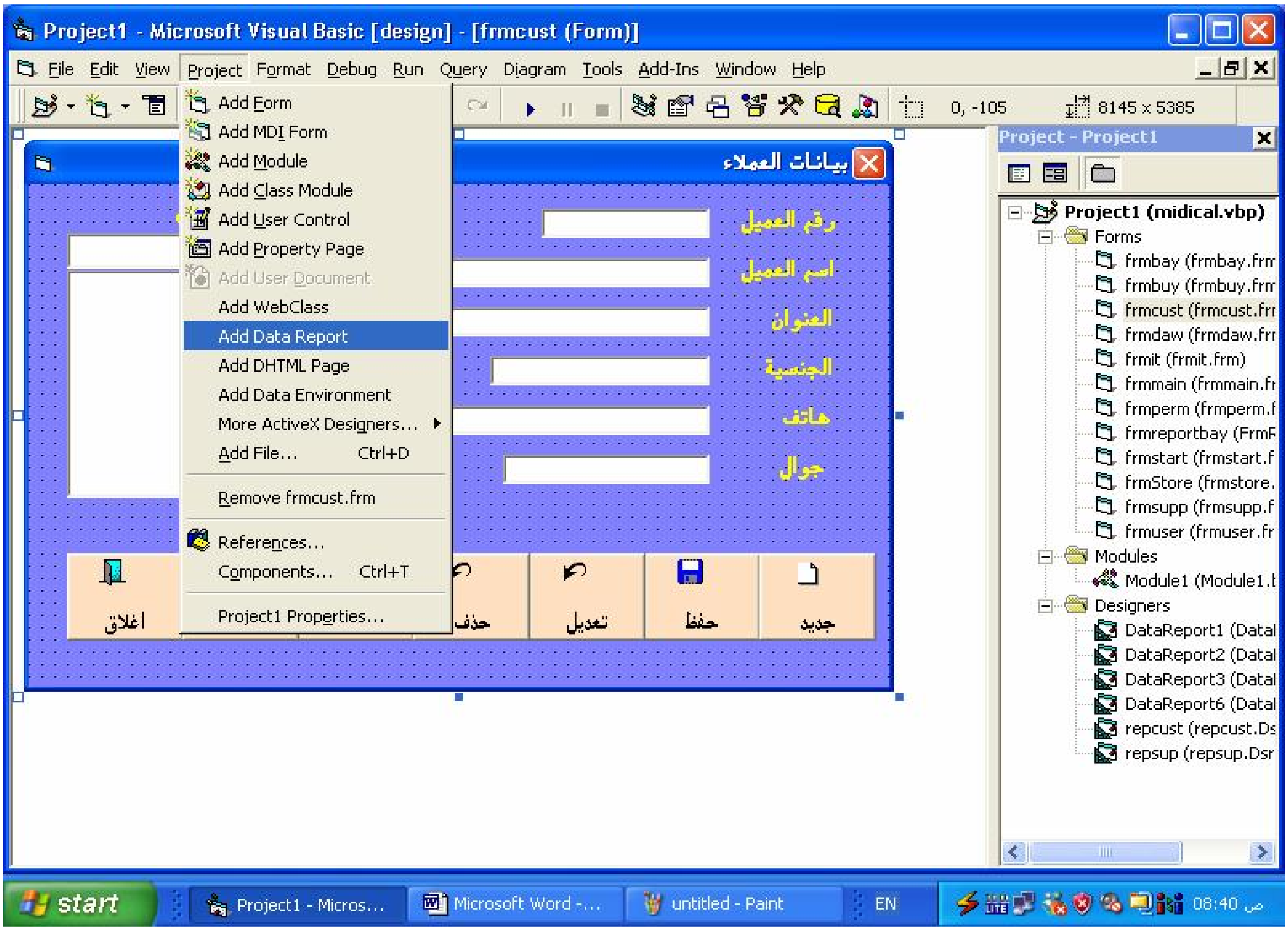
كيفية استخدام DATA REPORT مع الفيجوال بيسك ٦,٠٠

خطوات التعامل مع التقارير Data Report

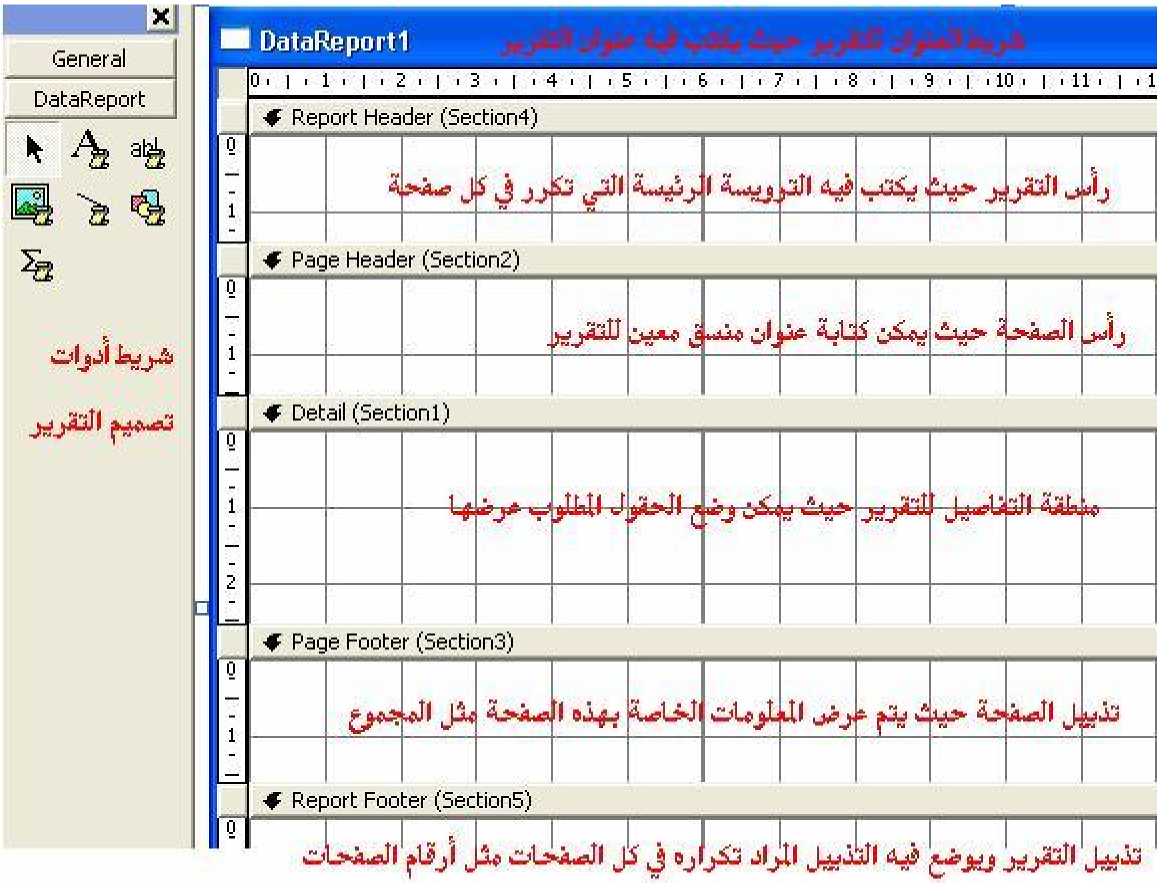
الخطوة الأولى : تصميم النموذج ووضع زر طباعة على النموذج مثل التصميم المرفق بالصورة التالية



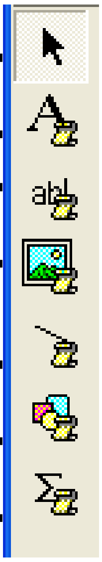
٢ – نبدأ بتصميم التقرير بحيث نختار قائمة project ثم add data report من القائمة المنسدلة كما هو واضح بالصورة



٣ – يظهر التقرير كما ترى بالصورة التالية



أدوات تصميم التقرير

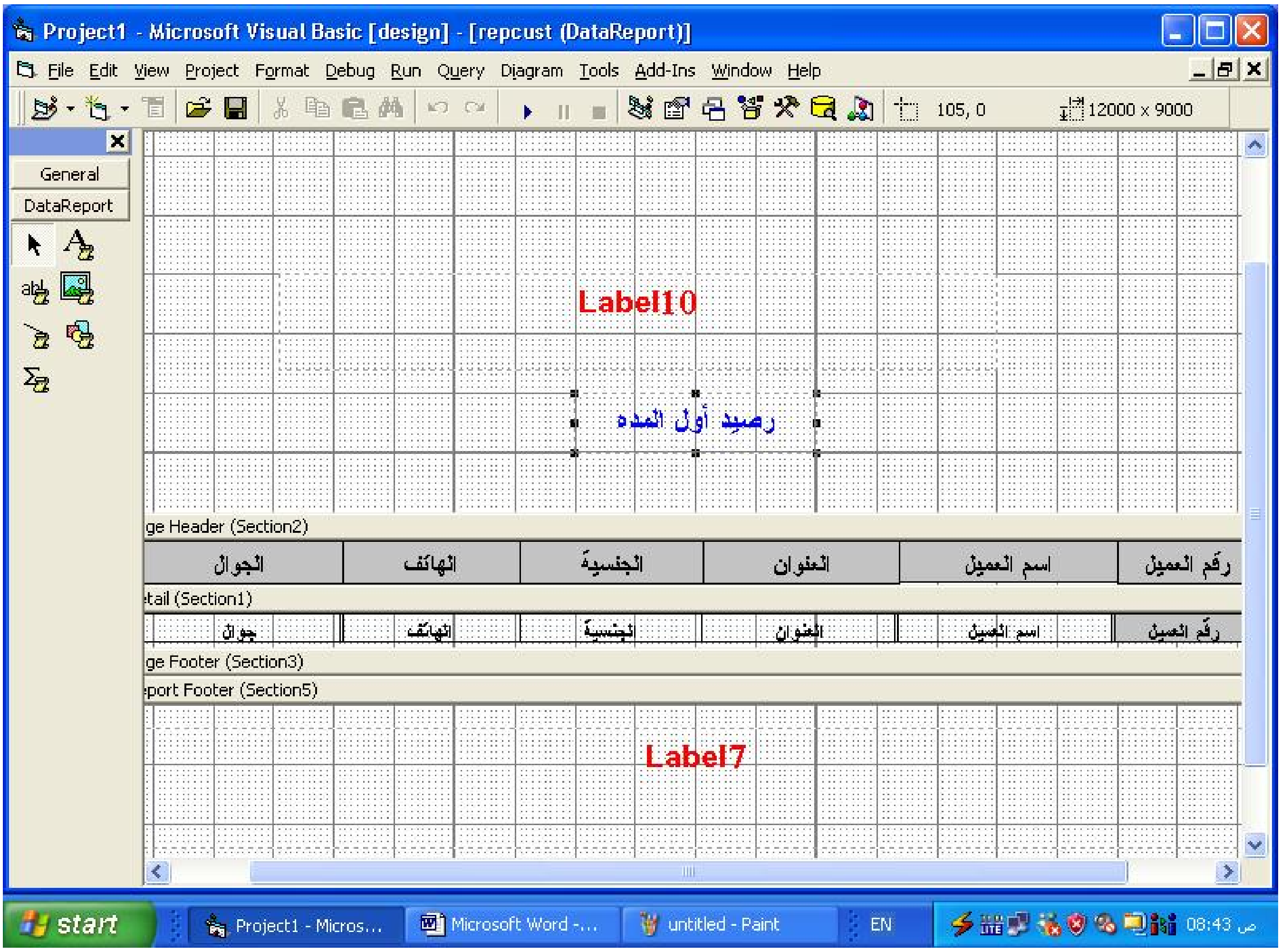


١ – إلغاء تحديد أي أداة
أداة العنوان حيث يمكن استخدامها في عمل عناوين الصفحات وعناوين الحقول
أداة مربع النص حيث توضع للتعبير عن عناوين الحقول من الجدول
إدراج صورة في التقرير
إدراج خط مستقيم
إدراج أشكال مثل المربع والدائرة والمستطيل
أداة جمع مجموعة من الحقول

بعض الخصائص الهامة لتصميم التقرير

اسم الخاصية	العمل
name	اسم التقرير وهو في حالتنا **repcust**
caption	عنوان التقرير وهو في حالتنا بيانات العملاء
gridX	نقاط محور السينات ويفيد في زيادة عدد نقاط الشبكة ويفضل أن تكون ١٠
gridY	نقاط محور الصادات ويفيد في زيادة عدد نقاط الشبكة ويفضل أن تكون ١٠
Data field	حيث يكتب اسم الحقل المراد إظهاره

وضع الحقول والعناوين والصور على التقرير



حيث يتم كتابة اسم الحقل في خاصية Data Filed لمربعات النصوص المعبرة عن الحقول

كتابة التعليمات لإظهار البيانات المطلوبة :

طباعة كل البيانات في الجدول

إذا أردنا طباعة كل العملاء في جدول العملاء عميل محدد يتم الضغط على اسم العميل (حسب تصميمك لشاشة العملاء) ثم نضغط زر طباعة ويكون الكود كالتالي

```
If rsrep.State = adStateOpen Then rsrep.Close
rsrep.Open "emp", db, adOpenDynamic, adLockOptimistic
Set repemp.DataSource = rsrep
  repemp.Sections("section2").Controls("lb1").Caption =
Format(Date, "yyyy/mm/dd("
  repemp.Show
dlist
```

طباعة موظف محدد

إذا أردنا طباعة عميل محدد يتم الضغط على اسم العميل (حسب تصميمك لشاشة العملاء) ثم نضغط زر طباعة ويكون الكود كالتالي

```
Dim sql As String
sql = "select * from emp where emp_no = " & Val(Text1.Text(
If rsrep.State = adStateOpen Then rsrep.Close
rsrep.Open sql, db, adOpenDynamic, adLockOptimistic
Set repemp.DataSource = rsrep
  repemp.Sections("section2").Controls("lb1").Caption =
Format(Date, "yyyy/mm/dd("
  repemp.Show
dlist
```

وبهذا نكون قد أنهينا هذه السلسة من الدروس لإنشاء مشروع كامل وأتمنى من الله تعالى أن تكونوا قد استفدتم من هذه الدروس وان تكون قد أضافت شيء لمعلوماتكم عن البرمجة باستخدام قواعد البيانات والى لقاء قادم عن شاء الله والسلام عليكم ورحمة الله وبركاته

تم الكتاب بحمد الله تعالي

جدة ١٤٢٦/٢/٤ الموافق ٢٠٠٥/٣/١٤ م